where $c_n = a_n + b_n$. For multiplication, we have

$$p(x) \cdot q(x) = d_0 + d_1 x + \cdots + d_n x^n + \cdots ,$$

where $d_n = \sum_{i=0}^{n} a_i b_{n-i}$. Although polynomial addition is obviously commutative we note that if R is not a commutative ring, then $\sum_{i=0}^{n} a_i b_{n-i}$ need not be equal to $\sum_{i=0}^{n} b_i a_{n-i}$. However, associativity of polynomial multiplication follows directly from the fact that $(R, \cdot)$ is a semigroup. In particular, suppose $p(x)$ and $q(x)$ are as above, $r(x) = \sum_{i=0}^{\infty} c_i x^i$, $d(x) = p(x) \cdot q(x)$, and $s(x) = q(x) \cdot r(x)$. Then

$$d(x) \cdot r(x) = \sum_{n=0}^{\infty} e_n x^n, \text{ where } e_n = \sum_{h=0}^{n} d_h c_{n-h} \text{ and } d_h = \sum_{i=0}^{h} a_i b_{h-i} .$$

Also,

$$s(x) = \sum_{m=0}^{\infty} s_m x^m, \ where\ s_m = \sum_{j=0}^{m} b_j c_{m-j} .$$

Therefore,

$$[p(x) \cdot q(x)] \cdot r(x) = \sum_{n=0}^{\infty} \left(\sum_{h=0}^{n} d_h c_{n-h} \right) x^n = \sum_{n=0}^{\infty} \left[\sum_{h=0}^{n} \left(\sum_{i=0}^{h} a_i b_{h-i} \right) c_{n-h} \right] x^n = \sum_{n=0}^{\infty} \left[\sum_{i+j+k=n} a_i b_j c_k \right] x^n$$

$$= \sum_{n=0}^{\infty} \left[\sum_{m=0}^{n} a_{n-m} \left(\sum_{j=0}^{m} b_j c_{m-j} \right) \right] x^n = \left(\sum_{i=0}^{\infty} a_i x^i \right) \left[\sum_{m=0}^{\infty} \left(\sum_{j=0}^{m} b_j c_{m-j} \right) x^m \right] = p(x) \cdot [q(x) \cdot r(x)] .$$

The fact that multiplication distributes over addition can also be proven in the same straightforward but cumbersome fashion. Thus, our discussion shows that the three ring axioms hold for polynomials with coefficients in a ring. We state this observation as a theorem.

3.6.2 Theorem. *The set $R[x]$ of all polynomials in an indeterminate x with coefficients in a ring R is a ring under polynomial addition and multiplication. If R is commutative, then $R[x]$ is also commutative, and if R has unity 1, then 1 is also the unity for $R[x]$.*

Thus, $\mathbb{Z}[x]$ is the ring of polynomials with integral coefficients, $\mathbb{R}[x]$ is the ring of polynomials with real coefficients, and $\mathbb{C}[x]$ is the ring of polynomials with complex coefficients. The elements of these rings are the polynomials one encounters in elementary algebra. A less common ring of polynomials in one variable is the ring $\mathbb{Z}_2[x]$. Here we have

$$(x + 1) + (x + 1) = (1 + 1)x + (1 + 1) = 0x + 0 = 0$$

and

$$(x + 1)^2 = (x + 1)(x + 1) = x^2 + (1 + 1)x + 1 = x^2 + 1 .$$

The most important polynomial domains arise when the coefficient ring is a field $\mathbb{F}$. The ring of polynomials $\mathbb{F}[x]$ over a field $\mathbb{F}$ has a number of properties which parallel those of the ring of integers $\mathbb{Z}$. For example, we know from elementary number theory that if m is a positive integer and n any other integer, then there exists unique integers q and r such that

$$n = mq + r \ \text{ and } \ 0 \leq r < m .$$

This is also known as the *division algorithm* for the integers. A similar result holds for elements of $\mathbb{F}[x]$.

3.6.3 **Theorem.** (Division Algorithm for $\mathbb{F}[x]$) *Let*

$$p(x) = a_0 + a_1 x + \cdots + a_n x^n$$

and

$$s(x) = b_0 + b_1 x + \cdots + b_m x^m$$

be two elements of $\mathbb{F}[x]$*, with* a_n *and* b_m *both nonzero elements of* $\mathbb{F}$ *and* $m > 0$*. Then there exist unique polynomials* $q(x)$ *and* $r(x)$ *in* $\mathbb{F}[x]$ *such that*

$$p(x) = s(x) \cdot q(x) + r(x)\,,$$

with the degree of $r(x)$ *less than* $m = degree\ s(x)$*.*

Proof: Consider the set $X = \{p(x) - s(x) \cdot t(x) : t(x) \in \mathbb{F}[x]\}$. Let $r(x) \in X$ such that for any other $r'(x) \in X$, degree $r(x) \leq r'(x)$. Then $r(x)$ is of form $r(x) = p(x) - s(x) \cdot q(x)$ for some $q(x) \in \mathbb{F}[x]$. Adding $s(x) \cdot q(x)$ to both sides of this equation we have that

$$p(x) = s(x) \cdot q(x) + r(x)\,.$$

Next we show that the degree of $r(x)$ is less than m. Suppose that $r(x) = \sum_{i=0}^{k} c_i x^i$ with $c_i \in \mathbb{F}$ and $c_k \neq 0$ if $k \neq 0$. If $k \geq m$, then

$$p(x) - q(x) \cdot s(x) - (c_k/b_m)x^{k-m} \cdot s(x) = r(x) - (c_k/b_m)x^{k-m} \cdot s(x)\,. \qquad \text{(I)}$$

Now let the left side of Equation I be represented by

$$r'(x) = r(x) - (c_k/b_m)x^{k-m} \cdot s(x) = r(x) - c_k\left[x^k + (b_{m-1}/b_m)x^{k-1} + \cdots + (b_0/b_m)x^{k-m}\right].$$

and let $u(x) = (b_{m-1}/b_m)x^{k-1} + \cdots + (b_0/b_m)x^{k-m}$. Then

$$r'(x) = c_{k-1}x^{k-1} + \cdots + c_0 - c_k u(x)$$

which is of degree strictly less than k. Furthermore, setting $t(x) = \left[q(x) - (c_k/b_m)x^{k-m}\right]$ and substituting t into the right side of Equation I we obtain $r'(x) = p(x) - s(x) \cdot t\,(x)$. Thus, $r'(x)$ is an element of X of degree less than $r(x)$, contradicting the fact that $r(x)$ was selected to have minimal degree. Therefore k must be less than m.

We have left to show that $q(x)$ and $r(x)$ are unique. Suppose that

$$p(x) = s(x) \cdot q(x) + r(x)$$

and

$$p(x) = s(x) \cdot q'(x) + r'(x)\,.$$

Then subtracting the second equation from the first, we have

$$s(x) \cdot \left[q(x) - q'(x)\right] = r'(x) - r(x)\,.$$

Since the degree of $r'(x) - r(x)$ is less than the degree of $s(x)$, we must have that $q(x) - q'(x) = 0$ or $q(x) = q'(x)$. But then we must have that $r'(x) - r(x) = 0$ or $r'(x) = r(x)$.

Q.E.D.

The polynomial $s(x)$ in Theorem 3.6.3 is called a *divisor* of $p(x)$ whenever $r(x)$ is the zero polynomial. The polynomials $q(x)$ and $r(x)$ can be computed by the same long division as used in high school for dividing real polynomials.

If $\mathbb{F}$ is a field or a commutative ring and $c \in \mathbb{F}$, then the function $\psi_c : \mathbb{F}[x] \to \mathbb{F}$ defined by

$$\psi_c\left(\sum_{i=0}^{n} a_i x^i\right) = \sum_{i=0}^{n} a_i c^i$$

for each $a_o + a_1 x + \cdots a_n x^n \in \mathbb{F}[x]$ is a homomorphism. We leave it to the reader to check that ψ_c preserves the field operations of addition and multiplication. It is common practice to let the symbol $p(c)$ denote the evaluation $\psi_c(p(x)) = a_0 + a_1 c + \cdots a_n c^n$. This notation provides for the following definition.

3.6.4 **Definition.** Let $\mathbb{F}$ be a commutative ring or field, $c \in \mathbb{F}$, and $p(x) \in \mathbb{F}[x]$. If $p(c) = 0$, then c is a *zero of* $p(x)$.

In terms of this definition, the problem of *solving a polynomial equation* is identical to that of *finding all the zeros of the corresponding polynomial*. Of course, the zeros of a polynomial are found in terms factoring the polynomial into products of lower degree polynomials.

3.6.5 **Corollary.** *An element $c \in \mathbb{F}$ is a zero of $p(x) \in \mathbb{F}[x]$ $\iff$ $x - c$ is a factor of $p(x)$ in $\mathbb{F}[x]$.*

Proof: Suppose that $c \in \mathbb{F}$ and $p(c) = 0$. Then by Theorem **3.6.3** $\exists\, q(x), r(x) \in \mathbb{F}[x]$ such that

$$p(x) = (x - c) \cdot q(x) + r(x)\,,$$

where the degree of $r(x)$ is less than 1. It follows that $r(x) = a$ for some $a \in \mathbb{F}$. Thus

$$p(x) = (x - c) \cdot q(x) + a\,.$$

Applying the homomorphism ψ_c, we obtain

$$0 = p(c) = 0 \cdot q(c) + a\,.$$

Hence, $a = 0$ and $p(x) = (x - c) \cdot q(x)$. Thus, $(x - c)$ is a factor of $p(x)$.
Conversely, if $(x - c)$ is a factor of $p(x)$, then applying the homomorphism ψ_c to $p(x) = (x - c) \cdot q(x)$, we obtain $p(c) = 0 \cdot q(c) = 0$.

Q.E.D.

Factorization of polynomials plays an important role in the optimization of certain image transforms (Chapter **5**). Similar to the factorization of numbers into prime numbers, polynomials can only be factored into prime components that cannot be factored any further. More precisely, a nonconstant polynomial $p(x) \in \mathbb{F}[x]$ is an *irreducible* or *prime polynomial in* $\mathbb{F}[x]$ if $p(x)$ cannot be expressed as a product $s(x) \cdot q(x)$ of two polynomials in $\mathbb{F}[x]$, both of lower degree than the degree of $p(x)$.

In addition to factorization into prime components, the ring of polynomials $\mathbb{F}[x]$ over a field $\mathbb{F}$ has many other properties which parallel those of the ring $\mathbb{Z}$ of integers. For example, the set $\mathbb{F}[x]$ may be partitioned by any polynomial $s(x) \in \mathbb{F}[x]$ of degree $n \geq 1$ into a ring

$$\mathbb{F}[x]/(s(x)) = \{[p(x)], [q(x)], \ldots\}$$

of equivalence classes just as $\mathbb{Z}$ was partitioned into the ring $\mathbb{Z}/(n) = \mathbb{Z}_n$. For any $p(x) \in \mathbb{F}[x]$ we define

$$[p(x)] = \{p(x) + t(x) \cdot s(x) : t(x) \in \mathbb{F}[x]\} .$$

Then $p(x) \in [p(x)]$ since the zero element of $\mathbb{F}$ is also an element of $\mathbb{F}[x]$, and $[p(x)] = [q(x)]$ if and only if $s(x)$ is a divisor of $p(x) - q(x)$. In analogy with arithmetic modulo n, we denote this last condition by $p(x) \equiv q(x)(mod\, s(x))$.

We now define addition and multiplication on these equivalence classes by

$$[p(x)] + [q(x)] = [p(x) + q(x)]$$

and

$$[p(x)] \cdot [q(x)] = [p(x) \cdot q(x)] ,$$

respectively, and leave it to the reader to convince himself of the following facts:

(1) Addition and multiplication are well defined operations on $\mathbb{F}[x]/s(x)$.
(2) $\mathbb{F}[x]/s(x)$ has [0] as zero element and [1] as unity, where 0 and 1 are the zero and unity of $\mathbb{F}$, respectively.
(3) $\mathbb{F}[x]/s(x)$ is a commutative ring with unity.

As for the elements of $\mathbb{Z}_n$, we let the polynomial $p(x)$ represent the equivalence class $[p(x)]$, keeping in mind that polynomial arithmetic is performed mod $s(x)$. For instance, if $\mathbb{R}[x]/\left(x^3 - 1\right)$ denotes the quotient ring of polynomials mod $x^3 - 1$, and $x^2 - 1$ and $x - 1$ are viewed as elements of $\mathbb{R}[x]/\left(x^3 - 1\right)$, then $\left(x^2 - 1\right) \cdot (x - 1) = x^2 - x$. In general, if $\mathbb{F}[x]/(x^n - 1)$ denotes the ring of polynomials mod $(x^n - 1)$, then we consider the elements of this ring to be polynomials on which multiplication is performed by replacing x^n with 1 wherever it appears rather than using the equivalence class notation [ ].

One reason for our emphasis of polynomial rings is the immediate connection between convolution of sequences of numbers and polynomial products. Convolutions of sequences are used in digital signal processing as the primary means of filtering. In image processing, convolutions can be used for such methods as edge detection, image enhancement, and template matching. A commonly used filter is the non-recursive finite impulse response filter. A finite impulse response filter is simply a tapped delay line in which the outputs at each stage are multiplied by fixed constants and then summed. In precise mathematical terms, let $s = \{s_i\}_{i \in \mathbb{Z}}$ and $f = \{f_i\}_{i \in \mathbb{Z}}$ be two sequences of numbers. We assume that $s_i = 0$ whenever $i < 0$ or $i > n$ for some fixed positive integer n. Similarly, $f_i = 0$ whenever $i < 0$ or $i > m$ for some $m \in \mathbb{Z}^+$. If the sequence *s* represents the sampled input signal and *f* the sequence of filter tap weights, then both input and filter sequences can be represented by the polynomials

$$s(x) = \sum_{i=0}^{n} s_i x^i \quad \text{and} \quad f(x) = \sum_{i=0}^{m} f_i x^i .$$

The integer m indicates the number of stages in the filter and is assumed to be much smaller than n which is assumed to be very large. The *linear convolution* of the two sequences is simply the polynomial product $p(x) = s(x) \cdot f(x)$ and the coefficients of $p(x)$ are called the *output sequence* of the convolution.

Another form of convolution is the *cyclic convolution*, which is closely related to the linear convolution. Here the input sequence $s = \{s_i\}_{i=0}^{n-1}$ and filter sequence $f = \{f_j\}_{j=0}^{n-1}$ are assumed to have the same length. The *cyclic convolution*, of the two sequences is defined as the polynomial product in $\mathbb{F}[x]/(x^n - 1)$; that is,

$$p(x) = s(x) \cdot f(x) \, mod \, (x^n - 1) = \sum_{k=1}^{n-1} \left(\sum_{(i+j) mod \, n = k} s_i f_j \right) x^k ,$$

where the coefficients of $p(x)$ form the output sequence.

Linear convolutions can be computed from cyclic convolutions by separating the input sequence into short sections. Each section is then individually processed as a cyclic convolution and properly merged with the other sections to produce a linear convolution. For example, assume $s = \{s_i\}_{i=0}^{n}$ is our input sequence and $f = \{f_j\}_{j=0}^{n}$ the desired filter sequence. Suppose that we wish to compute the linear convolution $s(x) \cdot f(x)$ in terms of a cyclic convolution on sections of length k, where $m < k$ and n is much larger than k. One method for achieving this is as follows. For $0 \leq i < k$ and $0 \leq j < n$ we first construct polynomials $s^{(0)}(x)$, $s^{(1)}(x), \ldots, s^{(j)}(x)$ having coefficients

$$s_i^{(0)} = s_i, \; s_i^{(1)} = s_{i+(k-m)}, \; s_i^{(2)} = s_{i+2(k-m)}, \; \ldots, s_i^{(j)} = s_{i+j(k-m)} .$$

Note that there is an overlap of m coefficients in the sequences $s^{(l)}$ and $s^{(l+1)}$.

To illustrate the construction of these polynomials, consider the case where $k = 3$ and $m = 2$. Then $0 \leq i < 3$ and

$$s^{(0)} = \{s_0, s_1, s_2\}, \;\; s^{(1)} = \{s_1, s_2, s_3\}, \;\; s^{(2)} = \{s_2, s_3, s_4\}, \; \ldots, \; etc.$$

Now suppose that $p(x) = \sum_{h=0}^{\infty} p_h x^h$ represents the linear convolution $p(x) = s(x) \cdot f(x)$ and that $p^{(h)}(x) = s^{(h)}(x) \cdot f(x) \, mod\big(x^k - 1\big)$ represents the hth cyclic convolution segment. Then, except for the first m coefficients, all the coefficients of $p(x)$ can be found among the coefficients of the polynomials $p^{(h)}(x)$, where $0 \leq h \leq j$. A quick check shows that these coefficients are as follows:

$$p_i^{(0)} = p_i \, , \;\; p_i^{(1)} = p_{i+(k-m)} \, , \;\; p_i^{(2)} = p_{i+2(k-m)} \, , \; \ldots \; , p_i^{(j)} = p_{i+j(k-m)} \, ,$$

where $m \leq i < k$. As can be ascertained from the next example, each cyclic convolution produces $k - m$ coefficients of the linear convolution, the remaining m coefficients are simply discarded.

3.6.6 Example: Suppose s denotes the input signal with degree $s(x) = n$ very large. If the filter is of length three with tap weights $f_0 = -1$, $f_1 = 0$, and $f_2 = 1$, then $f(x) = x^2 - 1$. The linear convolution $p(x) = s(x) \cdot f(x)$ is given by

$$p(x) = -s_0 + (-s_1)x + (s_0 - s_2)x^2 + (s_1 - s_3)x^3 + \; \ldots + (s_{n-2} - s_n)x^{n+2} .$$

Thus, $p_0 = -s_0$, $p_1 = -s_1$, $p_h = (s_{h-2} - s_h)$ for $h > 1$. Computing the linear convolution in terms of cyclic convolutions of length $m = 3$ by using the formulation $p^{(h)}(x) = s^{(h)}(x) \cdot f(x) \, mod \, \big(x^k - 1\big)$ we obtain

$$s^{(h)}(x) = s_h + s_{h+1}x + s_{h+2}x^2, \;\; \text{for} \;\; h = 0, 1, \ldots, n-2 .$$

Therefore, $p^{(h)}(x) = [s_h + s_{h+1}x + s_{h+2}x^2] \cdot \left(x^2 - 1\right) \; mod \; \left(x^3 - 1\right)$ and, hence,

$$p^{(h)}(x) = (s_{h+1} - s_h) + (s_{h+2} - s_{h+1})x + (s_h - s_{h+2})x^2 \,.$$

Since $m = 2$, the first two coefficients of the cyclic convolution are discarded and the remaining $k - m = 3 - 2 = 1$, which in this case is $(s_h - s_{h+2})$, matches all the coefficients of the linear convolution except for the first $m = 2$ coefficients, namely p_0 and p_1.

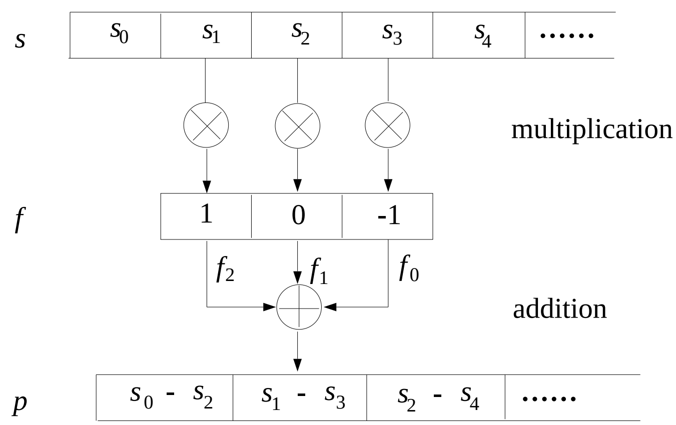

Figure 3.6.1 Linear convolution by sections.

Figure **3.6.1** provides a pictorial representation of the computation of a linear convolution as a convolution by sections. Here the filter is superimposed over part of the input sequence. After the necessary arithmetic computation is performed the filter is moved one step to the right and the computational process is repeated.

The reason for computing linear convolutions in terms of cyclic convolutions is the existence of a multitude of fast algorithms for computing short cyclic convolutions [5, 22].

The concept of a ring of polynomials in one variable can be generalized to rings of polynomials in several variables.

3.6.7 **Definition.** For any ring R, we define recursively

$$R[x_1, x_2, \ldots, x_n] = R[x_1, x_2, \ldots, x_{n-1}][x_n] \,.$$

If R is a ring with unity 1, then for any $r \in R$, we have that $xr = (1x)\left(rx^0\right)$. But by definition of polynomial multiplication, $(1x) \cdot \left(rx^0\right) = (1r)\left(xx^0\right) = rx$. Thus, $xr = rx$, which means that x commutes in $R[x]$ with any element of R. Similarly if R has unity, then so does $R[x]$ and, hence, in $R[x, y] = R[x][y]$ we have

$$yx = (1y) \cdot \left(x \cdot y^0\right) = (1x) \cdot \left(y \cdot y^0\right) = xy \,.$$

But then, clearly, $R[x][y] = R[y, x]$ or $R[x, y] = R[y, x]$. By repeated applications of this argument we see that in forming $R[x_1, x_2, \ldots x_n]$, the order in which the x_i's are adjoined is immaterial. It should be observed, however, that if R does not have unity, then it is meaningless to consider the product xr

in terms of the definitions of multiplication which we are using since, and this is the crux of the matter, x cannot be an element of $R[x]$. We shall not be too concerned with this subtlety as all rings we will use in our applications will be rings with unity.

As we have noted earlier, rings of polynomials in one variable provide a useful theoretical foundation for studying convolutions used in one-dimensional signal processing. A similar situation exists for studying convolutions for higher dimensional signal processing. For the analysis of 2–dimensional convolutions, which are a basic tool in digital image processing, the rings $\mathbb{F}[x, y]$ and $\mathbb{F}[x, y]/(x^m - 1, y^n - 1)$, where $\mathbb{F} \in \{\mathbb{R}, \mathbb{C}\}$, play a key role. Here $\mathbb{F}[x, y]/(x^m - 1, y^n - 1)$ denotes the quotient ring of polynomials mod $(x^m - 1)$ and mod $(y^n - 1)$; i.e., the elements of this ring are polynomials in two variables on which multiplication is performed by replacing x^m and y^n with 1 wherever they appear, rather than using sets of equivalence classes. As before, multiplication of elements in $\mathbb{F}[x, y]$ corresponds to linear convolutions, while multiplications of elements in $\mathbb{F}[x, y]/(x^m - 1, y^n - 1)$ corresponds to cyclic convolutions.

3.7 Vector Spaces

The theory of solutions of systems of linear equations is part of a more inclusive theory of an algebraic structure known as a "vector space." As we assume the reader's acquaintance with this topic, our treatment of vector spaces will be brief, designed only as a recall of basic concepts and theorems.

Although vector space theory as covered in elementary linear algebra courses is usually concerned with the Euclidean vector spaces $\mathbb{R}^n$, the operations of vector addition and scalar multiplication are used in many diverse contexts in mathematics and engineering. Regardless of the context, however, these operations obey the same set of arithmetic rules. In the general case, the scalars are elements of some field, which may be different than the real numbers.

3.7.1 **Definition.** A *vector space* $\mathbb{V}$ *over the field* $\mathbb{F}$, denoted by $\mathbb{V}(\mathbb{F})$, is an additive abelian group $\mathbb{V}$ together with an operation called *scalar multiplication* of each element of $\mathbb{V}$ by each element of $\mathbb{F}$ on the left, such that $\forall\, \alpha, \beta \in \mathbb{F}$ and $\mathbf{v}, \mathbf{w} \in \mathbb{V}$ the following five conditions are satisfied:

V_1 $\quad \alpha \cdot \mathbf{v} \in \mathbb{V}$

V_2 $\quad \alpha \cdot (\beta \cdot \mathbf{v}) = (\alpha\beta) \cdot \mathbf{v}$

V_3 $\quad (\alpha + \beta) \cdot \mathbf{v} = (\alpha \cdot \mathbf{v}) + (\beta \cdot \mathbf{v})$

V_4 $\quad \alpha \cdot (\mathbf{v} + \mathbf{w}) = (\alpha \cdot \mathbf{v}) + (\alpha \cdot \mathbf{w})$

V_5 $\quad 1 \cdot \mathbf{v} = \mathbf{v}$

The elements of $\mathbb{V}$ are called *vectors* and the elements of $\mathbb{F}$ are called *scalars*.

If the field $\mathbb{F}$ of scalars is clear from the context of discussion, then it is customary to use the symbol $\mathbb{V}$ instead of $\mathbb{V}(\mathbb{F})$. We also note that in comparison to the algebraic systems discussed thus far, multiplication for a vector space is not a binary operation on one set, but a rule which associates an element α from $\mathbb{F}$ and an element $\mathbf{v}$ from $\mathbb{V}$ with the element $\alpha \cdot \mathbf{v}$ of $\mathbb{V}$.

3.7.2 **Examples:**

(i) Consider the set $\mathbb{R}^X$. It follows directly from Theorem **2.8.2** that $(\mathbb{R}^X, +)$ is an abelian group which satisfies axioms V_1 through V_5 for all pairs of real numbers α and β. Thus $(\mathbb{R}^X, +)$ is a vector space over $\mathbb{R}$.

(ii) It follows from Example **3.5.6** that for any field $\mathbb{F}$, $M_{n\times n}(\mathbb{F})$ is an additive abelian group. It is also easy to see that if one defines scalar multiplication by

$$\alpha \cdot (a_{ij}) = (\alpha a_{ij}) \;\; \forall \alpha \in \mathbb{F} \;\text{ and }\; \forall (a_{ij}) \in M_n(\mathbb{F}),$$

then axioms V_1 through V_5 will hold. Thus, $M_{n\times n}(\mathbb{F})$ is a vector space over $\mathbb{F}$.

One of the most important concepts in the theory of vector spaces is the notion of linear independence.

3.7.3 **Definition.** Let $\mathbb{V}(\mathbb{F})$ be a vector space and $S = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ a subset of $\mathbb{V}$. If for every combination of scalars $\alpha_1, \alpha_2, \ldots, \alpha_k$

$$\alpha_1 \cdot \mathbf{v}_1 + \alpha_2 \cdot \mathbf{v}_2 + \ldots + \alpha_k \cdot \mathbf{v}_k = \mathbf{0} \quad \Rightarrow \quad \alpha_i = 0 \text{ for } i = 1, 2, \ldots, k,$$

then the vectors in S are said to be *linearly independent* over $\mathbb{F}$. In this definition, **0** denotes the zero vector in $\mathbb{V}$ and 0 denotes the zero of $\mathbb{F}$. If the vectors are not linearly independent over $\mathbb{F}$, then they are *linearly dependent* over $\mathbb{F}$.

Note that if the vectors are linearly dependent over $\mathbb{F}$, then for some combination of scalars $\alpha_1, \alpha_2, \ldots, \alpha_k \;\exists\; \alpha_i \neq 0$ for at least one $i = 1, 2, \ldots, k$ such that

$$\alpha_1 \cdot \mathbf{v_1} + \alpha \cdot \mathbf{v_2} + \ldots + \alpha_k \cdot \mathbf{v}_k = \mathbf{0}\,.$$

In this case we can solve for $\mathbf{v}_i$:

$$\mathbf{v}_i = \sum_{j=1\,(j\neq i)}^{k} \beta_j \mathbf{v}_j\,, \;\; \text{where } \beta_j = -\frac{\alpha_j}{\alpha_i}\,.$$

That is, $\mathbf{v}_i$ can be expressed as a linear combination of the remaining vectors; i.e., $\mathbf{v}_i$ *depends* on the remaining $\mathbf{v}_j$'s.

3.7.4 **Examples:**

(i) The vectors $(1,0), (0,1) \in \mathbb{R}^2$ are linearly independent (over $\mathbb{R}$) since

$$\alpha \cdot (1,0) + \beta \cdot (0,1) = (\alpha, \beta) = (0,0) \iff \alpha = 0 \text{ and } \beta = 0 \;\; \forall \alpha, \beta \in \mathbb{R}\,.$$

(ii) The vectors (-3,6), (3,-4), and (-1,0) are linearly dependent since

$$2 \cdot (-3,6) + 3 \cdot (3,-4) + 3 \cdot (-1,0) = (-6,12) + (9,-12) + (-3,0) = (0,0)\,.$$

(iii) The vectors

$$\mathbf{v} = \begin{pmatrix} 2 & 1/3 \\ 3 & 1 \end{pmatrix} \text{ and } \mathbf{w} = \begin{pmatrix} 6 & 1 \\ 9 & 3 \end{pmatrix}$$

are not linearly independent since $\mathbf{w} = 3\mathbf{v}$.

If $\mathbb{V}$ is a vector space, then certain subsets of $\mathbb{V}$ themselves form vector spaces under the vector addition and scalar multiplication defined on $\mathbb{V}$. These vector spaces are called *subspaces* of $\mathbb{V}$. For example, the set $\mathbb{W} = \{(x,y) \in \mathbb{R}^2 \,:\, y = 2x\}$ is a subspace of $\mathbb{R}^2$. Obviously, if $(x,y) \in \mathbb{W}$, then defining $u = \alpha x$ and $v = \alpha y$, we see that $v = \alpha y = \alpha(2x) = 2(\alpha x) = 2u$ for any real number α. Thus, $\alpha(x,y) = (\alpha x, \alpha y) = (u, 2u) \in \mathbb{W}$ and axiom V_1 is satisfied. The remaining four vector space axioms are just as easily verified. However, in order to show that a subset $\mathbb{W}$ is a subspace of a vector space $\mathbb{V}$, it is not necessary to verify axioms V_2 through V_5. The following theorem, which we state without proof, shows that in addition to axiom V_1, all we need to show is that $\mathbb{W}$ is closed under vector addition.

3.7.5 Theorem. *If $\mathbb{W}$ is a non-empty subset of a vector space $\mathbb{V}$ over $\mathbb{F}$, then $\mathbb{W}$ is a subspace of $\mathbb{V}$ $\Leftrightarrow$ the following conditions hold:*

(i) $\text{if } \mathbf{v}, \mathbf{w} \in \mathbb{W}, \text{ then } \mathbf{v} + \mathbf{w} \in \mathbb{W}$

(ii) $\alpha\mathbf{w} \in \mathbb{W} \;\forall \alpha \in \mathbb{F} \text{ and } \mathbf{w} \in \mathbb{W}$.

Let $\mathbb{W} = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ be a subset of a vector space $\mathbb{V}$ and let $S(\mathbb{W})$ denote the set of all linear combinations of the vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$. That is,

$$S(\mathbb{W}) = \left\{ \mathbf{v} \,:\, \mathbf{v} = \sum_{i=1}^{k} \alpha_i \mathbf{v}_i \right\}.$$

Then if $\mathbf{v}$, $\mathbf{w} \in S(\mathbb{W})$, with $\mathbf{v} = \sum_{i=1}^{k} \alpha_i \mathbf{v}_i$ and $\mathbf{w} = \sum_{i=1}^{k} \beta_i \mathbf{v}_i$, we have for $\alpha \in \mathbb{F}$ $\alpha\mathbf{v} = \sum_{i=1}^{k} \gamma_i \mathbf{v}_i$, where $\gamma_i = \alpha\alpha_i$, and $\mathbf{v} + \mathbf{w} = \sum_{i=1}^{k} \delta_i \mathbf{v}_i$ with $\delta_i = \alpha_i + \beta_i$. Therefore, $\alpha\mathbf{v}$ and $\mathbf{v} + \mathbf{w}$ are elements of $S(\mathbb{W})$ and by **3.7.5**, $S(\mathbb{W})$ is a subspace of $\mathbb{V}$. The subspace $S(\mathbb{W})$ of $\mathbb{V}$ is said to be *spanned* by $\mathbb{W}$ and the vectors of $\mathbb{W}$ are called the *generators* of the vector space $S(\mathbb{W})$. If $S(\mathbb{W}) = \mathbb{V}$, then we say that $\mathbb{W}$ is a *span* for $\mathbb{V}$. In particular, if $\mathbb{W}$ is a span for $\mathbb{V}$, then every vector $\mathbf{v} \in \mathbb{V}$ is a linear combination of the vectors in $\mathbb{W}$. For example, if $(x,y) \in \mathbb{R}^2$, then $(x,y) = x(1,0) + 0(1,1) + y(0,1)$. Thus any vector in $\mathbb{R}^2$ can be written as a linear combination of vectors from $\mathbb{W} = \{(1,0), (1,1), (0,1)\}$. Therefore, $S(\mathbb{W}) = \mathbb{R}^2$. Similarly, if $\mathbb{U} = \{(1,0), (1,1)\}$ and $(x,y) \in \mathbb{R}^2$, then $(x,y) = (x-y)(1,0) + y(1,1)$ and, therefore, $S(\mathbb{U}) = \mathbb{R}^2$. It follows that spans are not unique.

Closely related to the notion of a span are the concepts of *basis* and *dimension*. To students of science, the concept of dimension is a natural one. They usually think of a line as one-dimensional, a plane as two-dimensional, and the space around them as three-dimensional. The following definition makes these concepts more precise.

3.7.6 Definition. A *basis* for a vector space $\mathbb{V}$ is a linearly independent set of vectors of $\mathbb{V}$ which spans $\mathbb{V}$. If a basis for $\mathbb{V}$ contains only a finite number of vectors, then $\mathbb{V}$ is *finite dimensional*.

3.7.7 Example:

(i) Let $\mathbf{u}_1 = (1, 0, \ldots, 0)$, $\mathbf{u}_2 = (0, 1, 0, \ldots, 0)$, $\ldots$, $\mathbf{u}_n = (0, \ldots, 0, 1)$, then $\mathbb{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ is a basis for $\mathbb{R}^n$ since any vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ can be uniquely expressed as $\mathbf{x} = x_1\mathbf{u}_1 + x_2\mathbf{u}_2 + \cdots + x_n\mathbf{u}_n$. $\mathbb{U}$ is called the *standard basis* for $\mathbb{R}^n$, The dimension of $\mathbb{R}^n$ is *n*, the number of basis vectors.

(ii) Let

$$\mathbf{v}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \ \mathbf{v}_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \ \mathbf{v}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \ \text{and } \mathbf{v}_4 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Then $B = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\}$ is a basis for $M_{2\times 2}(\mathbb{R})$. Any vector $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ can be uniquely written as $\begin{pmatrix} a & b \\ c & d \end{pmatrix} = a\mathbf{v}_1 + b\mathbf{v}_2 + c\mathbf{v}_3 + d\mathbf{v}_4$. Note that $M_{2\times 2}(\mathbb{R})$ is of dimension four.

If $\mathbb{V}$ is a vector space with basis $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$, then any vector $\mathbf{v} \in \mathbb{V}$ can be uniquely expressed as a linear combination of the basis vectors $\mathbf{v} = \sum_{i=1}^{n} a_i\mathbf{v}_i$. It follows that $\mathbf{v}$ has a unique representation as an *n*-tuple vector $\mathbf{v} = (a_1, a_2, \ldots, a_n)$ since the scalars a_i express $\mathbf{v}$ uniquely in terms of a linear combination with respect to the basis $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$.

We remind the reader that a basis for a vector space is merely one of many kinds of bases encountered in studying various mathematical systems. A basis for a vector space portrays the algebraic properties of the vector space and is intimately connected with the *linear* algebraic properties of the space. Once a basis for a mathematical system has been established, we may proceed to describe the properties of the system under investigation relative to that basis. The particular form in which these properties manifest themselves may well depend upon what basis we chose to study them. The 2^n–topological space $\mathbb{Z}^n$ has different topological properties than the von Neumann space $\mathbb{Z}^n$. The most important facts about a vector space basis can be summarized as follows. Let $\mathbb{V}$ be a vector space and $B = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\} \subset \mathbb{V}$.

1. If B is a basis for $\mathbb{V}$, then every vector of $\mathbb{V}$ can be expressed *uniquely* as a linear combination of the elements of B.
2. If every vector of $\mathbb{V}$ can be uniquely expressed as a linear combination of the elements of B, then B is a basis for $\mathbb{V}$.
3. If B is a basis for $\mathbb{V}$ and $C \subset \mathbb{V}$ with $card(C) > card(B)$, then C is not linearly independent.

Statement 2. follows immediately from the observation that the zero vector has a unique representation in terms of the $\mathbf{v}_i$'s. Thus, the $\mathbf{v}_i$'s must be linearly independent. Statement 3. implies that the number of elements in a basis of a vector space must be unique; that is, *every* basis of a vector space has the same number of elements. With this in mind, we define the *dimension* of a vector space $\mathbb{V}$ as the cardinality of any basis for $\mathbb{V}$.

3.8 Linear Transformations

In the study of vector spaces the most important types of mappings are linear transformations. It should, therefore, come as no surprise that linear transformations from one vector space into another play an important role in image algebra which is concerned with the transformation of digital and continuous images.

3.8.1 **Definition.** A *linear transformation* or *linear operator* of a vector space $\mathbb{V}(\mathbb{F})$ into a vector space $\mathbb{W}(\mathbb{F})$ is a function $L : \mathbb{V}(\mathbb{F}) \to \mathbb{W}(\mathbb{F})$ which satisfies

$$L(\alpha \cdot \mathbf{u} + \beta \cdot \mathbf{v}) = \alpha \cdot L(\mathbf{u}) + \beta \cdot L(\mathbf{v})$$

for all $\mathbf{u}, \mathbf{v} \in \mathbb{V}(\mathbb{F})$ and for all scalars $\alpha, \beta \in \mathbb{F}$.

If in addition L is both one-to-one and onto, the L is called a *vector space isomorphism*. If $L : \mathbb{V}(\mathbb{F}) \to \mathbb{W}(\mathbb{F})$ is a vector space isomorphism, then $\mathbb{V}$ and $\mathbb{W}$ are said to be *isomorphic* vector spaces.

An equivalent definition of a linear transformation that can be found in many text books is given by the following theorem.

3.8.2 **Theorem.** $L : \mathbb{V}(\mathbb{F}) \to \mathbb{W}(\mathbb{F})$ *is a linear transformation* $\Leftrightarrow$ *for every* $\alpha \in \mathbb{F}$ *and for all* $\mathbf{u}, \mathbf{v} \in \mathbb{V}$
(i) $L(\alpha\mathbf{u}) = \alpha L(\mathbf{u})$ *and*
(ii)$L(\mathbf{u} + \mathbf{v}) = L(\mathbf{u}) + L(\mathbf{v})$

Proof: Suppose that L is a linear transformation. Then by setting either $\beta = 0$ or $\mathbf{v} = \mathbf{0}$ in Definition **3.8.1**, we obtain that $L(\alpha\mathbf{u}) = \alpha L(\mathbf{u})$. By setting $\alpha = 1 = \beta$, we obtain $L(\mathbf{u} + \mathbf{v}) = L(\mathbf{u}) + L(\mathbf{v})$.

Conversely, suppose that conditions (i) and (ii) hold. Let $\mathbf{u}_1 = \alpha\mathbf{u}$ and $\mathbf{v}_1 = \beta\mathbf{v}$. Then by condition (ii) we have that

$$(1) \;\; L(\alpha\mathbf{u} + \beta\mathbf{v}) = L(\mathbf{u}_1 + \mathbf{v}_1) = L(\mathbf{u}_1) + L(\mathbf{v}_1),$$

and from condition (i) we obtain

$$(2) \;\; L(\mathbf{u}_1) + L(\mathbf{v}_1) = L(\alpha\mathbf{u}) + L(\beta\mathbf{v}) = \alpha L(\mathbf{u}) + \beta L(\mathbf{v}).$$

Equations (1) and (2) imply

$$(3) \;\; L(\alpha\mathbf{u} + \beta\mathbf{v}) = \alpha L(\mathbf{u}) + \beta L(\mathbf{v}).$$

Thus L is a linear transformation.

Q.E.D.

If L is a linear transformation mapping a vector space $\mathbb{V}(\mathbb{F})$ into a vector space $\mathbb{W}(\mathbb{F})$, then
(i) $L(\mathbf{0}_{\mathbb{V}}) = \mathbf{0}_{\mathbb{W}}$, where $\mathbf{0}_{\mathbb{V}}$ and $\mathbf{0}_{\mathbb{W}}$ are the zero vectors in $\mathbb{V}$ and $\mathbb{W}$ respectively.
(ii) If $\mathbf{v}_1, \ldots, \mathbf{v}_n$ are elements of $\mathbb{V}$ and $\alpha_1, \ldots, \alpha_n$ are scalars from $\mathbb{F}$, then

$$L(\alpha_1 \cdot \mathbf{v}_1 + \alpha_2 \cdot \mathbf{v}_2 + \ldots + \alpha_n \cdot \mathbf{v}_n) = \alpha_1 \cdot L(\mathbf{v}_1) + \alpha_2 \cdot L(\mathbf{v}_2) + \ldots + \alpha_n \cdot L(\mathbf{v}_n)\,.$$

Statement (i) follows from the condition $L(\alpha \cdot \mathbf{u} + \beta \cdot \mathbf{v}) = \alpha \cdot L(\mathbf{u}) + \beta \cdot L(\mathbf{v})$ with $\alpha = 0$ and $\beta = 0$. Statement (ii) can be easily proven by mathematical induction.

3.8.3 Examples:

(i) Recall that the function $\nu : \mathbb{R}^X \to \mathbb{R}^n$ defined in Example **2.8.3** is one-to-one and onto whenever $X = \{1, 2, \ldots, n\}$. Now suppose that $k, k' \in \mathbb{R}$ and $f, g \in \mathbb{R}^X$. Then

$$\begin{aligned}
&\nu\big(k \cdot f + k' \cdot g\big) = \big(\big[k \cdot f + k' \cdot g\big](1),\ \big[k \cdot f + k' \cdot g\big](2),\ \ldots,\ \big[k \cdot f + k' \cdot g\big](n)\big) \\
&= \big(k \cdot f(1) + k' \cdot g(1),\ k \cdot f(2) + k' \cdot g(2),\ \ldots,\ k \cdot f(n) + k' \cdot g(n)\big) \\
&= (k \cdot f(1),\ k \cdot f(2),\ \ldots,\ k \cdot f(n)) + (k' \cdot g(1),\ k' \cdot g(2),\ \ldots,\ k' \cdot g(n)) \\
&= k \cdot (f(1),\ f(2),\ \ldots,\ f(n)) + k' \cdot (g(1),\ g(2),\ \ldots,\ g(n)) \\
&= k \cdot \nu(f(1),\ f(2),\ \ldots,\ f(n)) + k' \cdot \nu(g(1),\ g(2),\ \ldots,\ g(n)) \\
&= k \cdot \nu(f) + k' \cdot \nu(g)\,.
\end{aligned}$$

Thus, ν is a vector space isomorphism, and $\mathbb{R}^X$ and $\mathbb{R}^n$ are isomorphic vector spaces.

(ii) Let $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ and define $L : \mathbb{R}^2 \to \mathbb{R}^3$ by

$$L(\mathbf{x}) = (x_2, x_1, x_1 + x_2)\,.$$

Then for $\mathbf{y} = (y_1, y_2) \in \mathbb{R}^2$ and $\alpha, \beta \in \mathbb{R}$

$$\begin{aligned}
&L(\alpha \cdot \mathbf{x} + \beta \cdot \mathbf{y}) = L[(\alpha x_1, \alpha x_2) + (\beta y_1, \beta y_2)] = L(\alpha x_1 + \beta y_1, \alpha x_2 + \beta y_2) \\
&= (\alpha x_2 + \beta y_2,\ \alpha x_1 + \beta y_1,\ \alpha x_1 + \beta y_1 + \alpha x_2 + \beta y_2\ ) \\
&= (\alpha x_2,\ \alpha x_1,\ \alpha x_1 + \alpha x_2) + (\beta y_2,\ \beta y_1,\ \beta y_1 +\ \beta y_2\ ) \\
&= \alpha \cdot (x_2,\ x_1,\ x_1 + x_2) + \beta \cdot (y_2,\ y_1,\ y_1 + y_2) = \alpha \cdot L(\mathbf{x}) + \beta \cdot L(\mathbf{y})\,.
\end{aligned}$$

Note that if we define the matrix **a** by

$$\mathbf{a} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix},$$

then

$$L(\mathbf{x}) = \mathbf{x} \times \mathbf{a}$$

for each $\mathbf{x} \in \mathbb{R}^{\mathbf{2}}$. In general, if **a** is any $m \times n$ real valued matrix, then we can always define a linear operator $L_{\mathbf{a}} : \mathbb{R}^m \to \mathbb{R}^n$ by

$$L_{\mathbf{a}}(\mathbf{x}) = \mathbf{x} \times \mathbf{a}$$

for each $\mathbf{x} \in \mathbb{R}^m$. The operator $L_{\mathbf{a}}$ is linear since

$$\begin{aligned}
&L_{\mathbf{a}}(\alpha \cdot \mathbf{x} + \beta \cdot \mathbf{y}) = (\alpha \cdot \mathbf{x} + \beta \cdot \mathbf{y}) \times \mathbf{a} \\
&= (\alpha \cdot \mathbf{x}) \times \mathbf{a} + (\beta \cdot \mathbf{y}) \times \mathbf{a} = \alpha \cdot (\mathbf{x} \times \mathbf{a}) + \beta \cdot (\mathbf{y} \times \mathbf{a}) \\
&= \alpha \cdot L_{\mathbf{a}}(\mathbf{x}) + \beta \cdot L_{\mathbf{a}}(\mathbf{y})\,.
\end{aligned}$$

Thus, we can think of each $m \times n$ real valued matrix **a** as a linear operator from $\mathbb{R}^m$ to $\mathbb{R}^n$.

(iii) Consider the mapping $L : \mathbb{R}^2 \to \mathbb{R}^2$ defined by $L(\mathbf{x}) = \big(x_1, x_2^2\big)$. Now

$$\begin{aligned}
&L(\mathbf{x} + \mathbf{y}) = L[(x_1, x_2) + (y_1, y_2)] \\
&= L(x_1 + y_1, x_2 + y_2) = \big(x_1 + y_1, (x_2 + y_2)^2\big)\,,
\end{aligned}$$

while

$$L(\mathbf{x}) + L(\mathbf{y}) = \big(x_1, x_2^2\big) + \big(y_1, y_2^2\big) = \big(x_1 + y_1, x_2^2 + y_2^2\big)\,.$$

Thus, $L(\mathbf{x} + \mathbf{y}) \neq L(\mathbf{x}) + L(\mathbf{y})$ and L is not a linear operator.

(iv) Suppose that $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ is a basis for $\mathbb{V}$ and $T : \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\} \rightarrow \mathbb{V}$ is a function defined by $T : \mathbf{v}_i \rightarrow \mathbf{u}_i$ for some elements $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ of $\mathbb{V}$. Then T defines a linear transformation, again denoted by T, of $\mathbb{V}$ into itself. The linear transformation is simply a *linear* extension of the original function defined as follows. Let $\mathbf{u} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i$ and $\mathbf{v} = \sum_{i=1}^{n} \beta_i \mathbf{v}_i$,. Now extend T by defining

$$(1)\ \ T : \alpha\mathbf{v} = \alpha \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \rightarrow \alpha \sum_{i=1}^{n} \alpha_i \mathbf{u}_i$$

and

$$(2)\ \ T : \mathbf{u} + \mathbf{v} = \sum_{i=1}^{n} (\alpha_i + \beta_i)\mathbf{v}_i \rightarrow \sum_{i=1}^{n} \alpha_i \mathbf{u}_i \ + \ \sum_{i=1}^{n} \beta_i \mathbf{u}_i \, .$$

Equations (1) and (2) imply that $T(\alpha\mathbf{u}) = \alpha T(\mathbf{u})$ and $T(\mathbf{u} + \mathbf{v}) = T(\mathbf{u}) + T(\mathbf{v})$. Hence, according to Theorem **3.8.2**, T is a linear transformation.

From Example **3.8.3**(iv) one can infer that any linear transformation of a vector space into itself can be described completely by exhibiting its effect on the basis of the space.

Suppose $T : \mathbb{V} \rightarrow \mathbb{W}$ is a linear transformation. Then the set $\{\mathbf{v} \in \mathbb{V} : T(\mathbf{v}) = \mathbf{0}\}$ is called the *kernel* of T and is denoted by $ker(T)$. Since $T(\mathbf{0}) = T(0 \cdot \mathbf{v}) = 0 \cdot T(\mathbf{v}) = \mathbf{0}$, $\mathbf{0} \in ker(T)$. Also, if $\mathbf{u}, \mathbf{v} \in ker(T)$, then $T(\mathbf{u} + \mathbf{v}) = T(\mathbf{u}) + T(\mathbf{v}) \ = \mathbf{0} + \mathbf{0} = \mathbf{0}$, and $T(k\mathbf{v}) = kT(\mathbf{v}) = k \cdot \mathbf{0} = \mathbf{0}$. Hence $\mathbf{u} + \mathbf{v}$ and $k\mathbf{v}$ are elements of $ker(T)$. According to **3.7.5**, $ker(T)$ is a subspace of $\mathbb{V}$. In a similar fashion one can show that $range(T)$ is a subspace of $\mathbb{W}$. We state these observations as a theorem.

3.8.4 Theorem. *If $T : \mathbb{V} \rightarrow \mathbb{W}$ is a linear transformation, then*

(a) *the kernel of T is a subspace of $\mathbb{V}$, and*

(b) *the range of T is a subspace of $\mathbb{W}$.*

The dimension of $range(T)$ is called the *rank* of T, and the dimension of the kernel of T is called the *nullity* of T. The important relationship between rank and nullity is given by the following theorem known as *the Dimension Theorem.*

3.8.5 Theorem. *If $T : \mathbb{V} \rightarrow \mathbb{W}$ is a linear transformation from an n-dimensional vector space $\mathbb{V}$ to a vector space $\mathbb{W}$, then*

$$rank(T) + nullity(T) = n \, .$$

A linear transformation $T : \mathbb{V} \rightarrow \mathbb{W}$ is *nonsingular* if and only if it is invertible; i.e., if and only if there exists a linear transformation $T^{-1} : range(T) \rightarrow \mathbb{V}$ such that $T^{-1}T = I$, where I denotes the identity transformation $I(\mathbf{v}) = I$ on $\mathbb{V}$.

The next theorem establishes the relationships of the notions of nonsingular, dimension, nullity, and rank.

3.8.6 **Theorem.** *Let $T : \mathbb{V} \to \mathbb{W}$ be a linear transformation from an n-dimensional vector space $\mathbb{V}$ to an n-dimensional vector space $\mathbb{W}$. Then the following are equivalent:*

(a) T *is nonsingular*

(b) T *is one-to-one*

(c) $ker(T) = \{\mathbf{0}\}$ *and* $range(T) = \mathbb{W}$

(d) $nullity(T) = 0$ *and* $rank(T) = n$

Proof: The equivalence of (a) and (b) follows immediately from Theorem **2.5.12**.

(b) $\Rightarrow$ (c). Suppose that $\mathbf{v} \neq \mathbf{0}$ and $\mathbf{v} \in ker(T)$. Then $T(\mathbf{v}) = \mathbf{0}$ and, hence, $\mathbf{0} = T^{-1}(\mathbf{0}) = T^{-1}T(\mathbf{v}) = I(\mathbf{v}) = \mathbf{v}$, which is a contradiction. Therefore, $ker(T) = \{\mathbf{0}\}$. Moreover, T being invertible implies that T is onto. Thus, $range(T) = \mathbb{W}$.

(c) $\Rightarrow$ (b). If $T(\mathbf{v}) = \mathbf{0}$, then $\mathbf{v} = \mathbf{0}$. Thus, when $T(\mathbf{u}) = T(\mathbf{v})$, we must have that $T(\mathbf{v} - \mathbf{u}) = \mathbf{0}$, which means that $\mathbf{v} = \mathbf{u}$. Therefore T is one-to-one. Also, if $range(T) = \mathbb{W}$, then T is onto.

(c) $\Rightarrow$ (a). This is obvious since we already have shown that (c) $\Rightarrow$ *T* is a one-to-one correspondence.

(c) $\Leftrightarrow$ (d). This follows immediately from the definition of *nullity* and *rank.*

Q.E.D.

This theorem implies the following corollary:

3.8.7 **Corollary.** *Let $T : \mathbb{V} \to \mathbb{W}$. Then T is one-to-one* $\Leftrightarrow$ $ker(T) = \{\mathbf{0}\}$.

Nonsingular linear transformations have the important property of mapping basis into basis.

3.8.8 **Theorem.** *If T is a nonsingular transformation from $\mathbb{V} \to \mathbb{W}$, then T maps any basis for $\mathbb{V}$ into a basis for $\mathbb{W}$.*

Proof: Let $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ be a basis for $\mathbb{V}$. Then

$$T\left(\sum_{i=1}^{n} a_i\mathbf{v}_i\right) = \sum_{i=1}^{n} a_iT(\mathbf{v}_i) \in range(T) = \mathbb{W}\,.$$

Clearly, $\{T(\mathbf{v}_1), T(\mathbf{v}_2), \ldots, T(\mathbf{v}_n)\}$ is a span of $\mathbb{W}$. Now suppose there exist constants $b_i \not= 0$ such that

$$\sum_{i=1}^{n} b_iT(\mathbf{v}_i) = \mathbf{0}.$$

Then

$$T^{-1}\left(\sum_{i=1}^{n} b_iT(\mathbf{v}_i)\right) = \sum_{i=1}^{n} b_iT^{-1}T(\mathbf{v}_i) = \sum_{i=1}^{n} b_i\mathbf{v}_i = T^{-1}(\mathbf{0}) = \mathbf{0}.$$

But this is a contradiction since it means that the vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ are linearly dependent. Hence $\{T(\mathbf{v}_1), T(\mathbf{v}_2), \ldots, T(\mathbf{v}_n)\}$ are linearly independent and form a basis for $\mathbb{W}$.

Q.E.D.

Composition of linear transformations provides the fundamental operation for the *algebra* of linear transformations. Of particular importance is the subalgebra of nonsingular transformations. Composition of nonsingular transformations is closed in the following sense:

3.8.9 **Theorem.** *Suppose $T : \mathbb{V} \to \mathbb{W}$ and $S : range(T) \to \mathbb{U}$. Then $S \circ T : \mathbb{V} \to \mathbb{U}$ is nonsingular if and only if S and T are nonsingular. Moreover, if $S \circ T$ is nonsingular, then*

$$(S \circ T)^{-1} = T^{-1} \circ S^{-1}.$$

Proof: If S and T are nonsingular, then clearly

$$(T^{-1} \circ S^{-1}) \circ (S \circ T) = T^{-1} \circ \left(S^{-1} \circ S\right) \circ T = T^{-1} \circ I \circ T = T^{-1} \circ T = I\,.$$

Thus, $S \circ T$ has an inverse, namely $(S \circ T)^{-1} = T^{-1} \circ S^{-1}$ and is, therefore, invertible. Conversely, suppose that $S \circ T$ is nonsingular. Then $(S \circ T)^{-1}$ exists and

$$S \circ \left[T \circ (S \circ T)^{-1}\right] = (S \circ T) \circ (S \circ T)^{-1} = I$$

and, hence, $T \circ (S \circ T)^{-1} = S^{-1}$. Thus, S is invertible. Therefore,

$$T \circ [(T \circ S)^{-1} \circ S] = [T \circ (T \circ S)^{-1}] \circ S = S^{-1} \circ S = I$$

which means that $T^{-1} = (T \circ S)^{-1} \circ S$. Hence, T is invertible.

Q.E.D.

As an easy consequence of this theorem we have

3.8.10 **Corollary.** *If $T : \mathbb{V} \to \mathbb{W}$ is nonsingular, then T^{-1} is nonsingular and $\left(T^{-1}\right)^{-1} = T$.*

3.8.11 **Example:** Let $T : \mathbb{R}^2 \to \mathbb{R}^2$ be defined by $T(x, y) = (\bar{x}, \bar{y})$, where

$$\bar{x} = x \cos\theta + y \sin\theta\,, \quad \bar{y} = -x \sin\theta + y \cos\theta$$

and θ some fixed angle. Then T rotates every point $(x, y) \in \mathbb{R}^2$ through an angle θ about the origin as shown (Figure **3.8.1**).

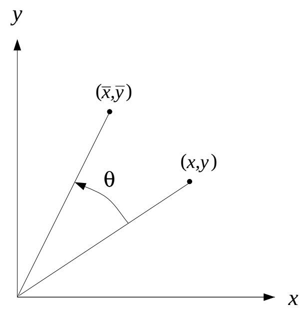

Figure 3.8.1 Planar rotation through an angle θ

Since

$$[(kx)\cos\theta + (ky)\sin\theta,\ -(kx)\sin\theta + (ky)\cos\theta] = k(x\cos\theta + y\sin\theta,\ -x\sin\theta + y\cos\theta),$$

we have that $T[k(x,y)] = k(\bar{x},\bar{y}) = kT(x,y)$. Furthermore,

$$\begin{aligned}
&T[(x_1,y_1)+(x_2,y_2)] = T(x_1+x_2, y_1+y_2)\\
&= [(x_1+x_2)\cos\theta + (y_1+y_2)\sin\theta,\ -(x_1+x_2)\sin\theta + (y_1+y_2)\cos\theta]\\
&= [(x_1\cos\theta + y_1\sin\theta) + (x_2\cos\theta + y_2\sin\theta), (-x_1\sin\theta + y_1\cos\theta) + (-x_2\sin\theta + y_2\cos\theta)]\\
&= (x_1\cos\theta + y_1\sin\theta,\ -x_1\sin\theta + y_1\cos\theta) + (x_2\cos\theta + y_2\sin\theta, -x_2\sin\theta + y_2\cos\theta)\\
&= (\bar{x}_1,\bar{y}_1) + (\bar{x}_2,\bar{y}_2) = T(x_1,y_1) + T(x_2,y_2).
\end{aligned}$$

This shows that T is a linear transformation.
Defining $S : \mathbb{R}^2 \to \mathbb{R}^2$ by $S(\bar{x},\bar{y}) = (x,y)$, where

$$x = \bar{x}\cos\theta - \bar{y}\sin\theta \quad and \quad y = \bar{x}\sin\theta + \bar{y}\cos\theta\,,$$

and using the above method of proof, it is easy to show that S is also a linear transformation.
To find the composition $S \circ T$, we compute

$$\begin{aligned}
&(S\circ T)(x,y) = S(T(x,y)) = S(\bar{x},\bar{y}) = (\bar{x}\cos\theta - \bar{y} sin\theta, \bar{x}\sin\theta + \bar{y}\cos\theta)\\
&= [(x\cos\theta + y\sin\theta)\cos\theta - (-x\sin\theta + y\cos\theta)\sin\theta,\\
&= (x\cos\theta + y\sin\theta)\sin\theta + (-x\sin\theta + y\cos\theta)\cos\theta]\\
&= \left(x\cos^2\theta + y\sin^2\theta, y\sin^2\theta + y\cos^2\theta\right) = \left[x\left(\cos^2\theta + \sin^2\theta\right), y\left(\sin^2\theta + \cos^2\theta\right)\right] = (x,y).
\end{aligned}$$

Therefore, $S \circ T = I$ and, hence, $S = T^{-1}$. Thus T is nonsingular.

In Example **3.8.3**, we observed that any real valued $m \times n$ matrix corresponds to a linear transformation $\mathbb{R}^m \to \mathbb{R}^n$. This observation also holds for $m \times n$ matrices with entries from any field $\mathbb{F}$ and linear transformations $\mathbb{V}(\mathbb{F}) \to \mathbb{W}(\mathbb{F})$, where $\mathbb{V}$ and $\mathbb{W}$ are vector spaces over $\mathbb{F}$, respectively. Suppose, for example, that $\mathbb{V}$ has basis $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m\}$, $\mathbb{W}$ has basis $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n\}$ and that $T : \mathbb{V} \to \mathbb{W}$ is a linear transformation. Then for each $i = 1, 2, \ldots, m$, $T(\mathbf{v}_i)$ can be expressed as a linear combination of the basis of $\mathbb{W}$: $T(\mathbf{v}_i) = t_{i1}\mathbf{w}_1 + t_{i2}\mathbf{w}_2 + \cdots + t_{in}\mathbf{w}_n$, where $t_{ij} \in \mathbb{F}$ for each $j = 1, 2, \ldots, n$. Now let $\mathbf{v} = a_1\mathbf{v}_1 + \cdots + a_m\mathbf{v}_m$ be an arbitrary vector of $\mathbb{V}$ and let $T(\mathbf{v}) = \mathbf{w}$. Then

$$\begin{aligned}
T(\mathbf{v}) = T\left(\sum_{i=1}^{m} a_i\mathbf{v}_i\right) &= \sum_{i=1}^{m} a_i T(\mathbf{v}_i) = \sum_{i=1}^{m} a_i \left(\sum_{j=1}^{n} t_{ij}\mathbf{w}_j\right) \\
&= \left(\sum_{i=1}^{m} a_i t_{i1}\right)\mathbf{w}_1 + \left(\sum_{i=1}^{m} a_i t_{i2}\right)\mathbf{w}_2 + \cdots + \left(\sum_{i=1}^{m} a_i t_{in}\right)\mathbf{w}_n = \mathbf{w}.
\end{aligned}$$

Therefore $\mathbf{w} = \left(\sum_{i=1}^{m} a_i t_{i1}, \sum_{i=1}^{m} a_i t_{i2}, \ldots, \sum_{i=1}^{m} a_i t_{in}\right)$. But this suggests that the transformation $T : \mathbb{V} \to \mathbb{W}$ is nothing more than the matrix product

$$(a_1, \ldots, a_m)\begin{pmatrix} t_{11} & t_{12} & \cdots & t_{1n} \\ t_{21} & t_{22} & \cdots & t_{2n} \\ \vdots & \vdots & & \vdots \\ t_{m1} & t_{m2} & \cdots & t_{mn} \end{pmatrix} = (b_1, \ldots, b_n), \tag{3.8.1}$$

where $b_j = \sum_{i=1}^{m} a_i t_{ij}$ for $j = 1, 2, \ldots, n$, and multiplication and addition are the appropriate field operations in $\mathbb{F}$.

It follows that the transformation T can be represented by the matrix (t_{ij}). Of course, the representation of T will be different for different bases of $\mathbb{V}$ and $\mathbb{W}$ as the vectors **v** and **w** are represented in terms of linear combinations of the basis vectors. If we changed one of the basis, then the question is "What is the matrix representing T relative to the new basis?" The answer to this question is closely linked to the notion of *equivalent* and *similar* transformations.

When two vector spaces are isomorphic, they are algebraically the same, and we are free to choose the simplest one in which to perform algebraic operations or solve algebraic problems. This observation applies just as well to all the other algebraic systems used in this treatise. It is particularly pertinent when solving linear image transform problems. It is therefore natural to inquire as to when two linear transformations on vector spaces are essentially the same.

Consider two linear transformations

$$S : \mathbb{U} \to \mathbb{V} \text{ and } T : \mathbb{W} \to \mathbb{X}.$$

Suppose that $\mathbb{U}$ and $\mathbb{W}$ are isomorphic and that $\mathbb{V}$ and $\mathbb{X}$ are also isomorphic. Let

$$P : \mathbb{U} \to \mathbb{W} \text{ and } Q : \mathbb{V} \to \mathbb{X}.$$

denote the respective isomorphisms. Since P and Q are one-to-one and onto functions, they have inverses

$$P^{-1} : \mathbb{W} \to \mathbb{U} \text{ and } Q^{-1} : \mathbb{X} \to \mathbb{V}.$$

Symbolically, this provides for the following diagram:

$$\begin{array}{ccc}
\mathbb{U} & \xrightarrow{\quad S \quad} & \mathbb{V} \\
P \downarrow\uparrow P^{-1} & & Q \downarrow\uparrow Q^{-1} \\
\mathbb{W} & \xrightarrow[\quad T \quad]{} & \mathbb{X}
\end{array}$$

Obviously the functions $\hat{S} = Q^{-1}TP : \mathbb{U} \to \mathbb{V}$ and $\hat{T} = QSP^{-1} : \mathbb{W} \to \mathbb{X}$ are linear transformations. Here we follow the convention of writing the composition of two transforms in juxtaposition so that $TP = T \circ P$. If $S = \hat{S}$ and $T = \hat{T}$, then S and T are *algebraically the same*; i.e., $S(\mathbf{u}) = \left(Q^{-1}TP\right)(\mathbf{u})$, which means that the result of computing $S(\mathbf{u})$ is exactly the same as the result obtained when computing $\left(Q^{-1}TP\right)(\mathbf{u})$. A similar observation holds for the computation of $T(\mathbf{w})$. If $S = \hat{S}$ and $T = \hat{T}$, then we say that the diagram *commutes*. We formulate the notion of algebraically similar as follows.

3.8.12 Definition. Two linear transformations

$$S : \mathbb{U} \to \mathbb{V} \text{ and } T : \mathbb{W} \to \mathbb{X}$$

are *isomorphically equivalent* if and only if there exists isomorphisms

$$P : \mathbb{U} \to \mathbb{W} \text{ and } Q : \mathbb{V} \to \mathbb{X}$$

such that $S = Q^{-1}TP$ and $T = QSP^{-1}$.
Two linear transformations

$$S : \mathbb{U} \to \mathbb{U} \text{ and } T : \mathbb{W} \to \mathbb{W}$$

are *similar* if and only if there exists an isomorphism

$$P : \mathbb{U} \to \mathbb{W}$$

such that

$$S = P^{-1}TP \text{ and } T = PSP^{-1}$$

.

Note that the notion of similar is simply a special case of isomorphically equivalent. Whenever $\mathbb{U} = \mathbb{V}$ and $\mathbb{W} = \mathbb{X}$, and S and T are isomorphically equivalent, then they are similar.

3.8.13 Example: Let $C\left(\mathbb{R}^1\right)$ denote the set of all complex valued continuous functions on $\mathbb{R}^1$. If $f \in C\left(\mathbb{R}^1\right)$, then f is said to *vanish at infinity* if $\forall\, \epsilon > 0$ there exists a compact set $K \subset \mathbb{R}^1$ such that $|f(x)| < \epsilon\ \forall x \notin K$. The *norm* of f is defined as

$$\|f\| = \int_{-\infty}^{\infty} |f(x)|dx.$$

Now let

$$\mathbb{U} = \left\{f \in C\left(\mathbb{R}^1\right) : \|f\| < \infty\right\} \quad \text{and}$$

$$\mathbb{V} = \left\{f \in C\left(\mathbb{R}^1\right) : f \text{ vanishes at infinity }\right\}$$

We leave it to the reader to convince himself that both $\mathbb{U}$ and $\mathbb{V}$ satisfy the axioms of a vector space. Continuous functions on $\mathbb{R}^1$ with finite norm are of special importance in signal processing since the convolution product of such functions is again a continuous function with finite norm. In particular, the *convolution product* $g = f * h$ is defined by

$$g(x) = \int_{-\infty}^{\infty} f(y)h(x-y)dy$$

The vector space $\mathbb{U}$ together with this product forms an interesting algebraic structure known as a *Banach algebra* [20].

Define the transforms $S : \mathbb{U} \to \mathbb{U}$ and $T : \mathbb{V} \to \mathbb{V}$ by

$$[S(f)](x) = \int_{-\infty}^{x} f(y)e^{-(x-y)}dy$$

and

$$[T(g)](y) = \frac{g(y)}{1+iy},$$

respectively.

The function $\mathcal{F} : \mathbb{U} \to \mathbb{V}$ defined by $\mathcal{F}(f) = \hat{f}$, where

$$\hat{f}(y) = \int_{-\infty}^{\infty} f(x)e^{-iyx}dx,$$

is called the *Fourier transform* of f. It is well known that the Fourier transform is one-to-one, onto, and preserves the vector space operations [20]. The inverse Fourier transform is given by $f = \mathcal{F}^{-1}\left(\hat{f}\right)$, where

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(y)e^{ixy}dy.$$

By defining

$$h(x) = \begin{cases} e^{-x} & if \;\; x > 0 \\ 0 & if \;\; x \le 0 \end{cases}$$

we see that

$$\int_{-\infty}^{x} f(y)e^{-(x-y)}dy = \int_{-\infty}^{\infty} f(y)h(x-y)dy.$$

Thus, $S(f)$ computes the convolution $S(f) = f * h$. Furthermore,

$$[\mathcal{F}(h)](y) = \hat{h}(y) = \int_{0}^{\infty} e^{-x}e^{-ixy}dx = \frac{1}{1+iy}.$$

Therefore, $\widehat{T(g)}$ is simply the product $\widehat{T(g)} = g \cdot \hat{h}$. It now follows from the convolution theorem $\widehat{g \cdot h} = g * h$ (see [20]) that

$$\left(\mathcal{F}^{-1} T \mathcal{F}\right)(f) = \left(\mathcal{F}^{-1} T\right)\left(\hat{f}\right) = \mathcal{F}^{-1}\left(\hat{f} \cdot \hat{h}\right) = \mathcal{F}^{-1}\left(\hat{f}\right) * \mathcal{F}^{-1}\left(\hat{h}\right) = f * h = S(f).$$

A similar argument shows that $\left(\mathcal{F} \mathcal{S} \mathcal{F}^{-1}\right) = T$. Thus, S and T are similar.

Since linear transforms can be expressed in terms of matrices, the use of terminology from matrix algebra is natural. Nonsingular transforms correspond to nonsingular matrices while similar transforms correspond to similar matrices. Recall that two n-square matrices A and B with entries from a field $\mathbb{F}$ are *similar* over $\mathbb{F}$ provided that there exists a nonsingular matrix P with entries from $\mathbb{F}$ such that $B = PAP^{-1}$. Eigenvalues and eigenvectors also have their analogous interpretation in linear transform theory. In particular, if $T : \mathbb{V} \to \mathbb{V}$ is a linear transform, then any vector $\mathbf{v} \in \mathbb{V}$ with the property that

$$T(\mathbf{v}) = \lambda \mathbf{v} \text{ for some scalar } \lambda$$

is called an *eigenvector* of T. The scalar λ is called the *eigenvalue* of T associated with the eigenvector $\mathbf{v}$.

Suppose that $\mathbf{T}$ is represented by the matrix $\mathbf{T} = (t_{ij})$, $\mathbf{v}$ is an eigenvector of $\mathbf{T}$ represented by a row vector $(v_1, \ldots, v_n)$, relative to a fixed basis, and λ is the associated eigenvalue. Then

$$T(\mathbf{v}) = \lambda \mathbf{v},$$

$$(v_1, \ldots, v_n)\mathbf{T} = \lambda(v_1, \ldots, v_n),$$

and

$$(v_1, \ldots, v_n)(\mathbf{T} - \lambda \mathbf{I}) = \mathbf{0},$$

where **I** denotes the *n*-square identity matrix and **0** the zero vector. This is the matrix form of a system of *n* linear homogeneous equations, and a nonzero solution $(v_1, \ldots, v_n)$ exists $\Leftrightarrow$ $\mathbf{T} - \lambda \mathbf{I}$ is singular. This occurs $\Leftrightarrow$ $det(\mathbf{T} - \lambda \mathbf{I}) = 0$. From our knowledge of determinants we see that

$$det(\mathbf{T} - \lambda \mathbf{I}) = \begin{vmatrix} t_{11} - \lambda & t_{12} & \ldots & t_{1n} \\ t_{21} & t_{22} - \lambda & \ldots & t_{2n} \\ \vdots & \vdots & & \vdots \\ t_{n1} & t_{n2} & \ldots & t_{nn} - \lambda \end{vmatrix} = 0$$

is a polynomial equation of degree *n* in λ, say

$$(-1)^n \lambda^n + b_1 \lambda^{n-1} + \cdots + b_{n-1}\lambda + b_n = 0,$$

where the b_i's are sums of products of the t_{ij}'s. The polynomial $det(\mathbf{T} - \lambda \mathbf{I})$ is called the *characteristic polynomial* of matrix $\mathbf{T}$ and also of the transform $\mathbf{T}$. The equation $det(\mathbf{T} - \lambda \mathbf{I}) = 0$ is called the *characteristic equation* of $\mathbf{T}$ and the roots of the characteristic equation are called the *eigenvalues* of the matrix $\mathbf{T}$.

Our discussion has established the following result:

3.8.14 Theorem. *The eigenvalues of a matrix* $\mathbf{A}$ *are the eigenvalues of the linear transformation represented by* $\mathbf{A}$ *in any coordinate system.*

3.8.15 Theorem. *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are similar, then* $\mathbf{A}$ *and* $\mathbf{B}$ *have the same characteristic polynomial and hence the same eigenvalues.*

We present two different proofs of this theorem, one formulated in the language of matrix theory and the other in the language of transformations.

Proof: (First method) If $\mathbf{A} = \mathbf{PBP}^{-1}$, then

$$\mathbf{A} - \lambda\mathbf{I} = \mathbf{PBP}^{-1} - \lambda\mathbf{I} = \mathbf{P}(\mathbf{B} - \lambda\mathbf{I})\mathbf{P}^{-1}$$
$$det(\mathbf{A} - \lambda\mathbf{I}) = (det\,\mathbf{P})det(\mathbf{B} - \lambda\mathbf{I})\big(det\,\mathbf{P}^{-1}\big) = det(\mathbf{B} - \lambda\mathbf{I}).$$

(Second method) If $\mathbf{A}$ and $\mathbf{B}$ are similar, then they represent the same transformation T relative to different bases. The eigenvalues of T depend only on T and are independent of the coordinate system (Theorem **3.8.14**). Hence $\mathbf{A}$ and $\mathbf{B}$ have the same eigenvalues. Therefore, their characteristic polynomials are alike except possibly for a multiplicative constant. Thus, $det(\mathbf{A} - \lambda\mathbf{I}) = k(\mathbf{B} - \lambda\mathbf{I})$ for all λ. For $\lambda = 0$ we have $det(\mathbf{A}) = k(det(\mathbf{B}))$, so that $k = 1$, since similar matrices have the same determinant.

Q.E.D.

We conclude this section with the well-known fact that the diagonal elements of a triangular matrix are its eigenvalues. The proof is straight forward and we leave it as an exercise to the reader.

3.8.16 Theorem. *The eigenvalues of a triangular matrix are the diagonal elements. In particular, the eigenvalues of a diagonal matrix are the diagonal elements.*

3.9 Linear Algebras

One way of viewing linear transformations is as operators that transform vectors of one space into vectors of the same space or a different space. New insight into the behavior of linear transformations can be gained by viewing them as *operands* instead of operators. If $\mathbb{L}$ denotes the set of all linear transformations of a vector space $\mathbb{V}(\mathbb{F})$ into itself, then we define an addition and multiplication on $\mathbb{L}$ by

$$T + S \;\;:\;\; (T + S)(\mathbf{v}) = T(\mathbf{v}) + S(\mathbf{v})\,,\;\; \mathbf{v} \in \mathbb{V}(\mathbb{F})$$

and

$$T \circ S \;\;:\;\; (T \circ S)(\mathbf{v}) = T(S(\mathbf{v}))\,,\;\; \mathbf{v} \in \mathbb{V}(\mathbb{F})$$

for all $S, T \in \mathbb{L}$.

These two operations endow $\mathbb{L}$ with a rich algebraic structure. First note that $(\mathbb{L}, +)$ is an abelian group. The additive zero is the transform $O \in \mathbb{L}$ defined by $O(\mathbf{v}) = \mathbf{0}$, where $\mathbf{0}$ denotes the zero vector in $\mathbb{V}$. If $T \in \mathbb{L}$, then its additive inverse is the transform $-T$ defined by $-T(\mathbf{v}) = -(T(\mathbf{v}))$. Also, for $R, S, T \in \mathbb{L}$ we have

$$(T + S)(\mathbf{v}) = T(\mathbf{v}) + S(\mathbf{v}) = S(\mathbf{v}) + T(\mathbf{v}) = (S + T)(\mathbf{v})$$

and

$$(R+(S+T))(\mathbf{v}) = R(\mathbf{v}) + (S+T)(\mathbf{v}) = R(\mathbf{v}) + (S(\mathbf{v}) + T(\mathbf{v}))$$
$$= (R(\mathbf{v}) + S(\mathbf{v})) + T(\mathbf{v}) = ((R+S)(\mathbf{v})) + T(\mathbf{v}) = ((R+S)+T)(\mathbf{v})\,.$$

This proves commutativity and associativity.

Defining scalar multiplication on $\mathbb{L}$ by

$$\alpha T \;:\; (\alpha T)(\mathbf{v}) = \alpha(T(\mathbf{v}))\,,\;\; \mathbf{v} \in \mathbb{V}(\mathbb{F})$$

for all $T \in \mathbb{L}$ and $\alpha \in \mathbb{F}$, we obtain

$$[\alpha(S+T)](\mathbf{v}) = \alpha[(S+T)(\mathbf{v})] = \alpha[S(\mathbf{v}) + T(\mathbf{v})]$$
$$= \alpha(S(\mathbf{v})) + \alpha(T(\mathbf{v})) = (\alpha S)(\mathbf{v}) + (\alpha T)(\mathbf{v}) = (\alpha S + \alpha T)(\mathbf{v})$$

and

$$[(\alpha+\beta)T](\mathbf{v}) = (\alpha+\beta)(T(\mathbf{v})) = \alpha(T(\mathbf{v})) + \beta(T(\mathbf{v})) = (\alpha T + \beta T)(\mathbf{v})\,.$$

This shows that

$$\alpha(S+T) = \alpha S + \alpha T \;\text{ and }\; (\alpha+\beta)T = \alpha T + \beta T\,.$$

These observations provide us with the next theorem.

3.9.1 Theorem. $\mathbb{L}$ *is a vector space over* $\mathbb{F}$.

In our observations we have not taken the multiplication $T \circ S$ into account. As remarked in the previous section, multiplication, which is defined as composition, is a fundamental operation in the algebra of linear transformations. The importance of this multiplication is established by the following fact.

3.9.2 Theorem. $(\mathbb{L}, +, \circ)$ *is a ring.*

Proof: We already know that $(\mathbb{L}, +)$ is an abelian group. Multiplication is clearly associative (but in general not commutative). To complete the proof that $\mathbb{L}$ is a ring we prove the left distributive law

$$T \circ (S+R) = (T \circ S) + (T \circ R)$$

and leave the right distributivity to the reader. We have

$$[T \circ (S+R)](\mathbf{v}) = T[(S+R)(\mathbf{v})] = T(S(\mathbf{v}) + R(\mathbf{v}))$$
$$= T(S(\mathbf{v})) + T(R(\mathbf{v})) = (T \circ S)(\mathbf{v}) + (T \circ R)(\mathbf{v}) = [(T \circ S) + (T \circ R)](\mathbf{v})\,.$$

Q.E.D.

Restricting our attention to non-singular transformations, we can make an even stronger case for the operation of multiplication. Let $\mathfrak{N} = \{T \in \mathbb{L} \,:\, T \text{ is non–singular}\}$.

3.9.3 **Theorem.** $(\mathfrak{N}, \circ)$ *is a group.*

Proof: It follows from Theorem **3.8.9** that $\mathfrak{N}$ is closed under multiplication. The associative law holds in $\mathfrak{N}$ since it holds in $\mathbb{L}$. Let $I \in \mathbb{L}$ be the identity map $I(\mathbf{v}) = \mathbf{v}$. Evidently I is non-singular and, hence, a member of $\mathfrak{N}$. It is the multiplicative identity since

$$(I \circ T)(\mathbf{v}) = I(T(\mathbf{v})) = T(\mathbf{v}) = T(I(\mathbf{v})) = (T \circ I)(\mathbf{v}).$$

Since each $T \in \mathfrak{N}$ is a one-to-one mapping, it has an inverse T^{-1} defined by $T^{-1}(T(\mathbf{v})) = \mathbf{v}$. Obviously, $\big(T^{-1} \circ T\big)(\mathbf{v}) = T^{-1}(T(\mathbf{v})) = \mathbf{v} = T\big(T^{-1}(\mathbf{v})\big) = \big(T \circ T^{-1}\big)(\mathbf{v})$. Also, the inverse of a one-to-one function is one-to-one and, hence, $T^{-1} \in \mathfrak{N}$. Therefore $\mathfrak{N}$ is a multiplicative group.

Q.E.D.

Vector spaces, linear transformations, and matrix operations are so vital in applications that they are usually studied under one parasol commonly known as *linear algebra*. In the previous section we observed that linear transformations and matrices are algebraically equivalent entities, while Theorem **3.9.1** shows that the set of linear transformations on a vector space is itself a vector space. These observations actually provide the basis for a formal definition of the somehow fuzzy concept of linear algebra. Specifically, a *linear algebra* over a field $\mathbb{F}$ is defined as a set $\mathfrak{L}$ having addition and multiplication, together with scalar multiplication by elements of $\mathbb{F}$ such that the following axioms are satisfied:

L_1 Under addition and scalar multiplication, $\mathfrak{L}$ is a vector space $\mathfrak{L}(\mathbb{F})$ over $\mathbb{F}$

L_2 Multiplication is associative

L_3 Multiplication is both left and right distributive over addition

L_4 $\mathfrak{L}$ has a multiplicative identity element

L_5 $(\alpha\cdot\mathbf{v}) * \mathrm{w} = \mathrm{v} * (\alpha\cdot\mathbf{w}) = \alpha\cdot(\mathbf{v} * \mathbf{w}) \;\; \forall\, \mathbf{v}, \mathbf{w} \in \mathfrak{L} \text{ and } \forall\, \alpha \in \mathbb{F}$

In axiom L_5 the symbol $\cdot$ corresponds to scalar multiplication and $*$ to the multiplication defined on $\mathfrak{L}$. It follows that a linear algebra $\mathfrak{L}$ over $\mathbb{F}$ consists of two sets together with five *distinct* operations

$$\mathfrak{L}(\mathbb{F}) = \{\mathfrak{L}, \mathbb{F}, +, \times, \dot{+}, \cdot, *\},$$

where $\{\mathfrak{L}, \mathbb{F}, +, \times, \dot{+}, \cdot\}$ is a vector space. Here $+$ and $\times$ denote the field operations of addition and multiplication on $\mathbb{F}$, $\dot{+}$ denotes vector addition in $\mathfrak{L}$, and $\cdot$ denotes multiplication of a vector by a scalar. Of course, we follow the conventional abuse of notation and let + denote vector as well as scalar addition and $\cdot$ denote both multiplication on $\mathbb{F}$ as well as scalar multiplication of a vector. Indeed, in many cases we shall even dispense with the symbol $\cdot$ entirely and simply write $\alpha\mathbf{v}$ in order to denote $\alpha \cdot \mathbf{v}$. This abuse of notation causes very little confusion as the type of operation an author has in mind is usually apparent from the context in which it is used.

3.9.5 **Examples:**

(i) The field $\mathbb{C}$ of complex numbers is a linear algebra of dimension 2 over the field $\mathbb{R}$ of real numbers. $\mathbb{C}(\mathbb{R})$ is a vector space of dimension 2 that satisfies axioms L_2 through L_5.

(ii) $\mathbb{L}(\mathbb{F})$ is a linear algebra. This follows from Theorems **3.9.1** and **3.9.2**. If $\mathbb{L}$ is the set of all linear transformations of a vector space $\mathbb{V}(\mathbb{F})$ of dimension n, then $\mathbb{L}(\mathbb{F})$ is of dimension n^2. Hence the algebra $M_n(\mathbb{F})$ of all $n \times n$ matrices over $\mathbb{F}$ is also a linear algebra.

(iii) An important example of a linear algebra of dimension four was given a century ago by Hamilton. The elements of this algebra are called *quaternions*, and the scalars are real numbers. Historically, the algebra of quaternions can be considered as the forerunner of matrix algebra. A quaternion is an expression of form

$$a_1\mathbf{1} + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k}.$$

The 4–tuple $(\mathbf{1}, \mathbf{i}, \mathbf{j}, \mathbf{k})$ is a basis element and $a_i \in \mathbb{R}$ for $i = 1, \ldots, 4$. Let Q denote the set all quaternions. As an additive group we may view Q as $(\mathbb{R}^4, +)$, where addition is defined component-wise. In particular, renaming the components of the basis elements by

$$\mathbf{1} = (1,0,0,0), \;\; \mathbf{i} = (0,1,0,0), \;\; \mathbf{j} = (0,0,1,0), \;\; \text{and} \;\; \mathbf{k} = (0,0,0,1)$$

and agreeing to let

$$a_1 = (a_1,0,0,0), \;\; a_2\mathbf{i} = (0,a_2,0,0), \;\; a_3\mathbf{j} = (0,0,a_3,0), \;\; \text{and} \;\; a_4\mathbf{k} = (0,0,0,a_4)\,,$$

then it follows from our definition that

$$(a_1, a_2, a_3, a_4) = a_1 + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k}\,.$$

Thus,

$$\begin{aligned}&(a_1 + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k}) + (b_1 + b_2\mathbf{i} + b_3\mathbf{j} + b_4\mathbf{k})\\ &= (a_1 + b_1) + (a_2 + b_2)\mathbf{i} + (a_3 + b_3)\mathbf{j} + (a_4 + b_4)\,\mathbf{k}\,.\end{aligned}$$

Defining scalar multiplication by

$$a \cdot (a_1 + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k}\,) = (aa_1) + (aa_2)\mathbf{i} + (aa_3)\mathbf{j} + (aa_4)\mathbf{k}$$

turns Q into a vector space over $\mathbb{R}$.

Multiplication on Q is defined in terms of the components of the basis elements:

$$\begin{aligned}&a\mathbf{1} = \mathbf{1}a = a \;\; \forall a \in Q, \;\; \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -\mathbf{1}, \;\; \text{and}\\ &\mathbf{ij} = \mathbf{k}, \;\; \mathbf{jk} = \mathbf{i}, \;\; \mathbf{ki} = \mathbf{j}, \;\; \mathbf{ji} = -\mathbf{k}, \;\; \mathbf{kj} = -\mathbf{i}, \;\; \text{and} \;\; \mathbf{ik} = -\mathbf{j}\,.\end{aligned}$$

To insure that the distributive law holds, we define

$$\begin{aligned}&(a_1 + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k})(b_1 + b_2\mathbf{i} + b_3\mathbf{j} + b_4\mathbf{k})\\ &= (a_1b_1 - a_2b_2 - a_3b_3 - a_4b_4) + (a_1b_2 + a_2b_1 + a_3b_4 - a_4b_3)\mathbf{i}\\ &+ (a_1b_3 - a_2b_4 + a_3b_1 + a_4b_2)\mathbf{j} + (a_1b_4 + a_2b_3 - a_3b_2 + a_4b_1)\mathbf{k}\,.\end{aligned}$$

Obviously, since $\mathbf{ij} = \mathbf{k} \neq -\mathbf{k} = \mathbf{ji}$, multiplication in Q is not commutative. However, if $a = a_1 + a_2\mathbf{i} + a_3\mathbf{j} + a_4\mathbf{k} \neq 0$, then

$$\frac{\bar{\mathbf{a}}}{|\mathbf{a}|} \cdot \mathbf{a} = \mathbf{1}, \;\; \text{where} \;\; \bar{\mathbf{a}} = a_1 - a_2\mathbf{i} - a_3\mathbf{j} - a_4\mathbf{k}\,, \;\; \text{and} \;\; |\mathbf{a}| = a_1^2 + a_2^2 + a_3^2 + a_4^2\,.$$

Therefore, every nonzero element of Q has a multiplicative inverse. From this we may conclude that Q is a non-commutative division ring that satisfies axioms L_1 through L_5. In 1878, Froboenius proved that Q is the *only* non-commutative linear division algebra over $\mathbb{R}$.

The linear algebra of Example **3.9.5**(ii) plays as important a role in the theory of linear algebra as the symmetric group S_n does in group theory. In Section **3.4** we remarked that every group of order n is isomorphic to some subgroup of S_n. Analogously, *any* linear algebra of dimension n is isomorphic to a subalgebra of $\mathbb{L}(\mathbb{F})$ (or $M_{n\times n}(\mathbb{F})$). This fact provides a concrete representation of any such abstract linear algebra and a striking illustration of the generality of linear transformations and their importance in linear algebra.

3.10 Group Algebras

Cyclic convolutions, which amount to polynomial multiplication (**3.6**), are a special case of multiplication in a group algebra. Paul D. Gader [12] and D. Wenzel [22] were the first to note the connection between group algebras, image algebra, and group algebra applications to image processing. In this section we briefly discuss some of the background of group algebras.

Let $G = \{g_\lambda : \lambda \in \Lambda\}$ be any multiplicative group and R be any commutative ring with unity. Let $R(G)$ denote the set of all *formal* sums of form

$$\sum_{\lambda\in\Lambda} r_\lambda g_\lambda ,$$

where $r_\lambda \in R, \;\; g_\lambda \in G, \;\; \text{and} \;\; r_\lambda = 0$ for all but a finite number of indices λ. If $\alpha \in R(G)$ with $\alpha = \Sigma_{\lambda\in\Lambda} r_\lambda g_\lambda$, then the r_λ's are called the coefficients of α. If $\beta \in R(G)$ with $\beta = \Sigma_{\lambda\in\Lambda} s_\lambda g_\lambda$, then we define addition by

$$\alpha + \beta = \sum_{\lambda\in\Lambda} (r_\lambda + s_\lambda) g_\lambda .$$

It is clear that $r_\lambda + s_\lambda = 0$ except for a finite number of indices λ, so $\alpha + \beta \in R(G)$. The additive identity is given by $0 = \Sigma_{\lambda\in\Lambda} 0 g_\lambda$. Since R is a commutative ring, we have as an immediate consequence that $(R(G), +)$ is an abelian group.

Multiplication of two elements of $R(G)$ is defined by

$$\alpha * \beta = \left(\sum_{\lambda\in\Lambda} r_\lambda g_\lambda\right)\left(\sum_{\lambda\in\Lambda} s_\lambda g_\lambda\right) = \sum_{\lambda\in\Lambda}\left(\sum_{g_\nu g_\xi = g_\lambda} r_\nu s_\xi\right) g_\lambda .$$

Observe that the product of two elements $g_\nu, g_\xi \in G$ is some element $g_\lambda = g_\nu g_\xi \in G$. Thus, naively, we formally distribute the sum $\Sigma_{\lambda\in\Lambda} r_\lambda g_\lambda$ over the sum $\Sigma_{\lambda\in\Lambda} s_\lambda g_\lambda$ and rename the term $r_\nu g_\nu s_\xi g_\xi$ by $r_\nu s_\xi g_\lambda$, where $g_\lambda = g_\nu g_\xi$ in G. Since r_ν and s_ξ are 0 for all but a finite number of λ's, the sum

$$\sum_{g_\nu g_\xi = g_\lambda} r_\nu s_\xi$$

contains only a finite number of nonzero summands $r_\nu s_\xi \in R$ and, hence, is an element of R. Clearly, again at most a finite number of such sums $\sum_{g_\nu g_\xi = g_\lambda} r_\nu s_\xi$ are nonzero. Therefore, $\alpha * \beta \in R(G)$.

The multiplication $*$ is called *group convolution* and the distributive laws of multiplication over addition follow at once from the definition of addition and multiplication. For the associativity of multiplication we have

$$\begin{aligned}
\left(\sum_{\lambda\in\Lambda} r_\lambda g_\lambda\right)\left[\left(\sum_{\lambda\in\Lambda} s_\lambda g_\lambda\right)\left(\sum_{\lambda\in\Lambda} t_\lambda g_\lambda\right)\right] &= \left(\sum_{\lambda\in\Lambda} r_\lambda g_\lambda\right)\left[\sum_{\lambda\in\Lambda}\left(\sum_{g_\nu g_\xi = g_\lambda} s_\nu t_\xi\right) g_\lambda\right] \\
&= \sum_{\lambda\in\Lambda}\left(\sum_{g_\mu g_\nu g_\xi = g_\lambda} r_\mu s_\nu t_\xi\right) g_\lambda \\
&= \left[\sum_{\lambda\in\Lambda}\left(\sum_{g_\mu g_\nu = g_\lambda} r_\mu s_\nu\right) g_\lambda\right]\left(\sum_{\lambda\in\Lambda} t_\lambda g_\lambda\right) \\
&= \left[\left(\sum_{\lambda\in\Lambda} r_\lambda g_\lambda\right)\left(\sum_{\lambda\in\Lambda} s_\lambda g_\lambda\right)\right]\left(\sum_{\lambda\in\Lambda} t_\lambda g_\lambda\right).
\end{aligned}$$

Thus, we have proven the following theorem.

3.10.1 Theorem. *If G is any multiplicative group, then $(R(G), +, *)$ is a ring which is commutative if and only if G is a commutative group.*

Rings have the two primitive operations necessary for defining the elements of $R(G)$. Any more specialized structure having these ring operations, such as a field, can of course be used as well.

3.10.2 Definition. The ring $R(G)$ defined above is called the *group ring of G over R*. If $\mathbb{F}$ is a field, then $\mathbb{F}(G)$ is called the *group algebra of G over $\mathbb{F}$*.

3.10.3 Examples:

(i) Let G be a cyclic group with generator x defined by the operation $x^i \cdot x^j = x^k$, where $i, j, k \in \mathbb{Z}$ and $k = i + j$. Then $G = \{ x^i : i \in \mathbb{Z} \}$ and

$$\mathbb{F}(G) = \left\{ \sum_{i\in\mathbb{Z}} r_i x^i \ : \ r_i = 0 \text{ for all but finitely many } i \in \mathbb{Z} \right\}.$$

The polynomials are a subset of this group algebra due to the constraint $r_i = 0$ for $i < 0$. The multiplication of this group algebra corresponds to linear convolutions when restricted to polynomials. Also note that G is isomorphic to $\mathbb{Z}$ and, therefore, $\mathbb{F}(G) \approx \mathbb{F}(\mathbb{Z})$.

(ii) Let $G_n = \{ x^i : i \in \mathbb{Z}_n \}$ with multiplication defined by $x^i \cdot y^j = x^{(i+j) mod\, n}$. Then cyclic convolution corresponds to multiplication in the group algebra $\mathbb{F}(G_n) \approx \mathbb{F}(\mathbb{Z}_n)$.

(iii) Let $G = \mathbb{Z}_2^2 = \{(0,0), (0,1), (1,0), (1,1)\}$ and let multiplication of elements of G be defined in terms of the exclusive or (xor); e.g., $(0,0)\cdot(0,0) = (0,0)$, $(1,0)\cdot(1,1) = (0,1)$, $(0,1)\cdot(0,0) = (0,1)$, etc. Renaming the elements of G by $g_0,\ g_1,\ g_2,$ and g_3, we obtain the group algebra

$$\mathbb{F}(G) = \mathbb{F}\big(\mathbb{Z}_2^2\big) = \left\{ \sum_{i\in\mathbb{Z}_4} r_i g_i \ :\ r_i \in \mathbb{F} \right\}.$$

More generally, let $G = \mathbb{Z}_2^m$ and suppose we have denoted the elements of G by $g_0, g_1, \ldots, g_{2^{m-1}}$. Then if $g_i = (i_0, \ldots, i_{m-1})$ and $g_j = (j_0, \ldots, j_{m-1})$, where $i_k,\ j_k \in \{0,1\}$ for $k = 0, 1, \ldots, m-1$, we define $g_i \cdot g_j = g_k \ \Leftrightarrow\ g_k = (i_0, \ldots, i_{m-1})\cdot(j_0, \ldots, j_{m-1})$. Again, multiplication is defined component-wise in terms of the usual exclusive or of binary numbers. The group convolution $*$ of the group algebra $\mathbb{R}(\mathbb{Z}_2^m)$ is called the *Walsh convolution.*

For each group element $\alpha \in \mathbb{F}(G)$ and each $c \in \mathbb{F}$ we define the scalar product $c\alpha$ by

$$c\alpha = c\left(\sum_{\lambda\in\Lambda} r_\lambda g_\lambda\right) = \sum_{\lambda\in\Lambda} (cr_\lambda) g_\lambda\,.$$

It is easily shown that this scalar multiplication satisfies axioms V_1 through V_5 (Section **3.7**). Thus, $(\mathbb{F}(G), +)$ together with scalar multiplication is a vector space over $\mathbb{F}$.

From our previous discussion we also know that $\mathbb{F}(G)$ satisfies axioms $L_1,\ L_2$, and L_3. Suppose 1 and e represent the multiplicative identities of $\mathbb{F}$ and G, respectively. Consider now the element $\iota \in \mathbb{F}(G)$ defined by

$$\iota = \sum_{\lambda\in\Lambda} r_\lambda g_\lambda\,,$$

where $r_\lambda = 0$ for all but one λ, say λ_0, and $r_{\lambda_0} = 1$ while $g_{\lambda_0} = e$. It then follows that

$$\alpha * \iota = \iota * \alpha = \alpha\ \ \forall\, \alpha \in \mathbb{F}(G)\,.$$

This shows that $\mathbb{F}(G)$ also satisfies axiom L_4. Axiom L_5 is a trivial consequence of the definitions of scalar multiplication and group convolution. This establishes the following

3.10.4 Theorem. *A group algebra together with scalar multiplication is a linear algebra.*

3.11 Lattice Algebra

The concept of lattices was formed with a view to generalize and unify certain relationships between subsets of a set, between substructures of an algebraic structure such as groups, and between geometric structures such as topological spaces. The development of the theory of lattices started about 1930 and was influenced by the work of Garrett Birkhoff [3].

Let Y be a partially ordered set with partial order $\preceq$. An *upper bound* (if it exists) of a subset X of Y is an element $y \in Y$ such that $x \preceq y \;\; \forall x \in X$. The *least upper bound* of X, denoted by $lubX$ or $supX$, is an upper bound $y_0 \in Y$ of X such that $y_0 \preceq y$ for every upper bound y of X. By the anti-symmetry property of partially ordered sets, $supX$ is unique if it exists. The notions of *lower bound* and *greatest lower bound*, denoted by $glbX$ or $infX$, are defined dually as in **2.10**. Again by anti-symmetry, $infX$ is unique if it exists.

3.11.1 **Definition.** A *lattice* is a partially ordered set L any two of whose elements x, y have a glb, denoted by $x \wedge y$, and a lub, denoted by $x \vee y$. A *sublattice* of a lattice L is a subset X of L, such that for each pair $x,\, y \in X$ we have that $x \vee y \in X$ and $x \wedge y \in X$. A lattice L is *complete* whenever for each of its subsets X, $infX$ and $supX$ exists.

Setting $X = L$, we see that any non-empty complete lattice L contains a least element O and a greatest element I. These elements are commonly called the *universal bounds* of L. It is also obvious that the dual of any lattice is a lattice, and that the dual of any complete lattice is a complete lattice with glb and lub interchanged.

3.11.2 **Examples:**

(i) In **3.1** we observed that $\mathbb{R}$ together with the relation of less or equal $(\leq)$ is a totally ordered set. If $x \vee y = max\{x,\, y\}$ and $x \wedge y = min\{x,\, y\} \;\; \forall x,\, y \in \mathbb{R}$, then $\mathbb{R}$ together with the operations of $\vee$ and $\wedge$ is a lattice. However, $(\mathbb{R},\, \vee,\, \wedge)$ is not a complete lattice as there is no largest and smallest number.

(ii) Let $\mathbb{R}_{\pm\infty} = \mathbb{R} \cup \{-\infty, \infty\}$, the set of real numbers with the symbols $-\infty$ and ∞ adjoined. Define $-\infty < x < \infty \;\; \forall x \in \mathbb{R}$ and $-\infty \leq x \leq \infty \;\; \forall x \in \{-\infty, \infty\}$. Then $(\mathbb{R}_{\pm\infty},\, \vee,\, \wedge)$ is a complete lattice with largest element ∞ and smallest element $-\infty$. The dual of this lattice is obtained by replacing $\leq$ with the relation of greater or equal $\geq$. Obviously, $(\mathbb{R},\, \vee,\, \wedge)$ is a sublattice of $(\mathbb{R}_{\pm\infty},\, \vee,\, \wedge)$.

(iii) The power set 2^X together with the order relation $\subset$ of set inclusion is a complete lattice with largest element X and smallest element $\varnothing$. For any family $\mathcal{A} = \{\mathcal{S}_\lambda \,:\, \lambda \in \Lambda\}$ of subsets of X, $inf\mathcal{A} = \bigcap\limits_{\lambda\in\Lambda} \mathcal{S}_\lambda$ and $sup\mathcal{A} = \bigcup\limits_{\lambda\in\Lambda} \mathcal{S}_\lambda$.

(iv) The set $\mathbb{R}^{\geq 0}_{\infty} = \mathbb{R}^{+} \cup \{0,\, \infty\}$ with the relation $\leq$ is a complete lattice. Here 0 is the smallest element and ∞ the largest element. $\left(\mathbb{R}^{\geq 0}_{\infty},\, \vee,\, \wedge\right)$ is a sublattice of $(\mathbb{R}_{\pm\infty},\, \vee,\, \wedge)$ but not of $(\mathbb{R},\, \vee,\, \wedge)$.

(v) We partially order $\mathbb{R}^{\mathbf{X}}$ by $f \leq g \;\Leftrightarrow\; f(x) \leq g(x) \;\; \forall x \in X$ and define $h = f \vee g$ by

$$h(x) = \begin{cases} f(x) & \text{if } g(x) \leq f(x) \\ g(x) & \text{if } f(x) < g(x) \end{cases}$$

and $k = f \wedge g$ by

$$k(x) = \begin{cases} f(x) & \text{if } f(x) \leq g(x) \\ g(x) & \text{if } g(x) < f(x)\,. \end{cases}$$

Then $(\mathbb{R}^{\mathbf{X}}, \vee, \wedge)$ is a lattice which is not complete. However, $(\mathbb{R}^{\mathbf{X}}_{\pm\infty}, \vee, \wedge)$ is a complete lattice with smallest element the constant function ϕ defined by $\phi(x) = -\infty \;\; \forall x \in X$ and largest element the constant function I defined by $I(x) = \infty \;\; \forall x \in X$.

The binary operations $\wedge$ and $\vee$ on lattices have several important properties, some of them being analogous to those of ordinary multiplication and addition. The following properties are easily verified [3].

3.11.3 Theorem. *If $(X, \preceq)$ is a partially ordered set, then the operations of $\vee$ and $\wedge$ satisfy the following equations (whenever the expressions referred to exist):*

(1) $x \wedge x = x, \quad x \vee x = x \qquad \text{(idempotent)}$

(2) $x \wedge y = y \wedge x, \quad x \vee y = y \vee x \qquad \text{(commutative)}$

(3) $x \wedge (y \wedge z) = (x \wedge y) \wedge z, \quad x \vee (y \vee z) = (x \vee y) \vee z \quad \text{(associative)}$

(4) $x \wedge (x \vee y) = x \vee (x \wedge y) = x \quad \text{(absorption)}$

(5) $x \preceq y \;\Leftrightarrow\; x \wedge y = x, \quad x \preceq y \;\Leftrightarrow\; x \vee y = y \quad \text{(consistency)}$

(6) *If X has a least element O, then $O \wedge x = O \;\; and \;\; O \vee x = x \;\; \forall x \in X$*

(7) *If X has a largest element I, then $I \wedge x = x \;\; and \;\; I \vee x = I \;\; \forall x \in X$*

Observe that in the above theorem we do not require that the partially ordered set be a lattice. In fact, Dedekind [11] used properties (1) through (4) to define lattices, and Birkhoff [3] proved that these properties completely characterize lattices. For lattices, we have the following additional algebraic relationships.

3.11.4 Theorem. *If $(L, \vee, \wedge)$ is a lattice with partial ordering $\preceq$, then*

(1) $y \preceq z \;\Rightarrow\; x \wedge y \preceq x \wedge z \;\; and \;\; x \vee y \preceq x \vee z \quad \text{(isotone)}$

(2) $x \preceq z \;\Rightarrow\; x \vee (y \wedge z) \preceq (x \vee y) \wedge z \qquad \text{(modularity)}$

(3) $x \wedge (y \vee z) \succeq \alpha(x \wedge y) \vee (x \wedge z) \qquad (\wedge \text{ distributivity})$

(4) $x \vee (y \wedge z) \preceq (x \vee y) \wedge (x \vee z) \qquad (\vee \text{ distributivity})$

Proof: We shall only prove property (2). The remaining properties are just as easy or follow directly from Theorem **3.11.3**.

(2) Obviously, $x \preceq x \vee y$ and, by hypothesis, $x \preceq z$. Hence, $x \preceq (x \vee y) \wedge z$. Also, $y \wedge z \preceq y \preceq x \vee y$ and $y \vee z \preceq z$. Thus, $y \wedge z \preceq (x \vee y) \wedge z$ and, hence, $x \vee (y \wedge z) \preceq (x \vee y) \wedge z$.
Q.E.D.

It is often convenient and much simpler to deal with only one of the operations of $\vee$ or $\wedge$ and obtain equivalent relations for the other through duality. For this, the following notion is helpful.

3.11.5 Definition. A *semilattice* is a commutative semigroup $(S, \bigcirc)$ which satisfies the idempotent law $x \bigcirc x = x$.

It follows from Theorem **3.11.3** that every partially ordered set X for which the operation $x \vee y$ is defined for each pair $x,\, y \in X$ is a semilattice. Conversely, under the relation defined by

$$x \preceq y \quad \Leftrightarrow \quad x \bigcirc y = y\,,$$

any semilattice with binary operation $\bigcirc$ becomes a partially ordered set in which $x \bigcirc y = lub\{x,\, y\}$.

3.11.6 Example: $(\mathbb{R},\, \vee)$ is a semilattice with dual $(\mathbb{R},\, \wedge)$. If $\mathbb{R}_{\infty} = \mathbb{R} \cup \{\infty\}$, and $\mathbb{R}_{-\infty} = \mathbb{R} \cup \{-\infty\}$, then $(\mathbb{R}_{-\infty}, \vee)$ is a semilattice with dual $(\mathbb{R}_{\infty}, \wedge)$. Note that the semilattice $(\mathbb{R}_{-\infty}, \vee)$ is a monoid with zero element $-\infty$ since $r \vee (-\infty) = (-\infty) \vee r = r \;\; \forall r \in \mathbb{R}_{-\infty}$. Similarly, $(\mathbb{R}_{\infty}, \wedge)$ is a monoid with zero element ∞.

As mentioned earlier, the operations of $\vee$ and $\wedge$ in a lattice are analogous to the arithmetic operations of + and $\times$. This analogy is most striking in distributive lattices.

3.11.7 Definition. A lattice $(L,\, \vee,\, \wedge)$ is a *distributive lattice* if and only if the following equality holds in L:

$(i) \qquad x \wedge (y \vee z) \;=\; (x \wedge y) \vee (x \wedge z).$

This equation expresses the similarity with the distributive law $x(y+z) \;=\; xy + xz$ of ordinary arithmetic. All lattices in Example **3.11.2** are distributive lattices. Note also that by duality we have that

$$(ii) \qquad x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$$

in any distributive lattice. Now if $x \preceq z$, then $z = x \vee z$. Thus, substituting *z* for $x \vee z$ in (ii) we have that any distributive lattice satisfies the following law:

$$(iii) \qquad x \preceq z \quad \Rightarrow \quad x \vee (y \wedge z) = (x \vee y) \wedge z\,.$$

Any lattice satisfying condition (iii) is called a *modular* lattice. Although every distributive lattice is modular, not every modular lattice is a distributive lattice.

3.11.8 Example: Let $L = \{O, a, b, c, I\}$ have partial order defined by $O \preceq a \preceq I,\;\; O \preceq b \preceq I$, and $O \;\preceq\; c \;\preceq\; I$. Then

$$a \wedge (b \vee c) = a \wedge I = a \neq O = O \vee O = (a \wedge b) \vee (a \wedge c)\,.$$

Thus L is not a distributive lattice. However, whenever $x \preceq z$ in L, then (iii) holds. For example, $a \preceq I$ and

$$a \vee (b \wedge I) = a \vee b = I = I \wedge I = (a \vee b) \wedge I\,.$$

Therefore, L is a modular lattice.

Suppose L is a lattice with smallest element *O* and largest element *I*, and $x \in L$. If $\exists\, x' \in L$ such that $x' \wedge x = O$ and $x' \vee x = I$, then x' is called the *complement* of x. Now if L is a distributive lattice and if $c \wedge x = c \wedge y$ and $c \vee x = c \vee y$, then $x = x \wedge (c \vee x) = x \wedge (c \vee y) = (x \wedge c) \vee (x \wedge y) = (c \wedge y) \vee (x \wedge y) = (c \vee x) \wedge y = (c \vee y) \wedge y = y$. Therefore, $x = y$. Thus, if x' and $\hat{x}$ are two complements of x so that $x \wedge x' = O = x \wedge \hat{x}$ and $x \vee x' = I = x \vee \hat{x}$, then by our analysis we have that $x' = \hat{x}$. That is, complements are unique in distributive lattices.

We say that a lattice is *complemented* if and only if all its elements have a complement.

Modular and complemented lattices are of special interest for applications to probability theory, including ergodic theory and multiplicative processes, to linear algebra, computer science, and engineering. For example, the set L of all subspaces of $\mathbb{R}^n$ is a complemented modular lattice. Here the orthogonal complement $S^\perp$ of any subspace S satisfies $S^\perp \cap S = \varnothing$ and $S^\perp \cup S = \mathbb{R}^n$. Also, by definition, a *Boolean lattice* is a complemented distributive lattice. The uses of Boolean algebras in computer science and engineering are manifold and range from the design of electrical networks to the theory of computing.

For Boolean lattices we have the following:

3.11.9 Theorem. *In any Boolean lattice, each element x has a unique complement x'. Furthermore, complementation satisfies the following equations*

(1) $\qquad x \wedge x' = O$ *and* $x \vee x' = I$

(2) $\qquad (x')' = x$

(3) $\qquad (x \wedge y)' = x' \vee y'$ *and* $(x \vee y)' = x' \wedge y'$

For a proof we refer the reader to [3]. Complementation should not be confused with conjugation or duality. For instance, if for every $r \in \mathbb{R}_{\pm\infty}$ we define its *dual* or *conjugate* r^* by $r^* = -r$, where $-(-\infty) = \infty$, then

$(2')$ $\qquad (r^*)^* = r$

$(3')$ $\qquad (r \wedge t)^* = r^* \vee t^*$ *and* $(r \vee t)^* = r^* \wedge t^*$

Equations $(2')$ and $(3')$ have the appearance of statements (2) and (3) of Theorem **3.11.9**. However, $r \wedge r^* = -\infty$ and $r \vee r^* = \infty$ if and only if $r = \infty$ or $r = -\infty$.

Since complements in a Boolean lattice L are unique, we can view complementation as a function $L \to L$ that maps $x \to x'$. From this point of view, complementation is a unary operation. If L is a Boolean lattice, then the algebra $(L, \vee, \wedge, ')$ determined by the two binary operations $\vee$ and $\wedge$, and the unary operation of complement on L is called a *Boolean algebra.*

3.11.10 Examples:

(i) It follows from the laws of operations on sets (**2.2.1**), that $\left(2^X, \cup, \cap, '\right)$ is a Boolean algebra.

(ii) Let $f \in \mathbb{Z}_2^X$ and define $h = f \vee g$ by

$$h(x) = \begin{cases} f(x) & \text{if } g(x) \leq f(x) \\ g(x) & \text{if } f(x) < g(x) \end{cases}$$

and $k = f \wedge g$ by

$$k(x) = \begin{cases} f(x) & \text{if } f(x) \leq g(x) \\ g(x) & \text{if } g(x) < f(x)\,. \end{cases}$$

Define complementation of f as the function $f' : X \to \mathbb{Z}_2$, where

$$f'(x) = \begin{cases} 1 & \text{if } f(x) = 0 \\ 0 & \text{if } f(x) = 1\,. \end{cases}$$

The least element is the zero function $O : X \to \mathbb{Z}_2$ defined as $O(x) = 0 \;\; \forall x \in X$ and the largest element is the constant function $I : X \to \mathbb{Z}_2$ defined by $I(x) = 1 \;\; \forall x \in X$. It is now easy to show that $\left(\mathbb{Z}_2^X, \vee, \wedge, {}'\right)$ is a Boolean algebra. This algebra provides a rigorous mathematical basis for the description of a wide range of Boolean image transformations.

(iii) The lattice $(\mathbb{R}_{\pm\infty}^{\mathbf{X}}, \vee, \wedge)$ of Example **3.11.2** (v) is a distributive lattice with minimal and maximal elements. Only the functions ϕ and I have complements, namely $I' = \phi$ and $\phi' = I$.

In addition to being a distributive lattice, the set of real numbers is also a ring, and our early experience in elementary algebra has taught us the useful properties

P_1 $\quad x \geq y \Rightarrow z + x \geq z + y$

P_2 $\quad x \geq 0 \;\text{ and }\; y \geq 0 \Rightarrow \; xy \geq 0$

P_3 $\quad z \geq 0 \Rightarrow z(x \vee y) = zx \vee zy \;\text{ and }\; z(x \wedge y) = zx \wedge zy\,,$

where $x, y, z \in \mathbb{R}$. These properties exhibit the interplay between the lattice and ring operations. The question naturally arises as to whether or not these properties hold in *any* ring (or semigroup, group, etc.) which is also a lattice. For the remainder of this section we will be mainly concerned with this question and examine properties of semigroup, groups, rings, and other algebraic structures which at the same time are lattices. While many properties can be couched in terms of lattice-related concepts, the main purpose of approaching these structures from a general algebraic viewpoint is to develop an anology to linear operator theory.

Suppose $\mathbb{F}$ is a set with binary operation $\times$ and partial order $\preceq$, and suppose that the system $(\mathbb{F}, \preceq, \times)$ satisfies the following property:

P_4 $\quad x \preceq y \Rightarrow a \times x \times b \preceq a \times y \times b \quad \forall a, b \in \mathbb{F}.$

If $(\mathbb{F}, \times)$ is a semigroup, then

1. $\mathbb{F}$ is called a *partially ordered semigroup* or *po-semigroup.*

2. If $(\mathbb{F}, \times)$ is a group, then $\mathbb{F}$ is called a *partially ordered group* or *po-group*.

3. If $(\mathbb{F}, \vee)$ is a semilattice and $(\mathbb{F}, \times)$ a semigroup, then $\mathbb{F}$ is called a *semilattice ordered semigroup* or *sl-semigroup.*

4. If $(\mathbb{F}, \vee)$ is a semilattice and $(\mathbb{F}, \times)$ a group, then $\mathbb{F}$ is called a *semilattice ordered group* or *sl-group*.

5. If $(\mathbb{F}, \vee, \wedge)$ is a lattice and $(\mathbb{F}, \times)$ a semigroup, then $\mathbb{F}$ is called a *lattice ordered semigroup* or *l-semigroup*.

6. If $(\mathbb{F}, \vee, \wedge)$ is a lattice and $(\mathbb{F}, \times)$ a group, then $\mathbb{F}$ is called a *lattice ordered group* or *l-group.*

Obviously, $(\mathbb{R}, \vee, \wedge, +)$ is an *l*-group since $P_1 \Rightarrow P_4$. On the other hand, $(\mathbb{R}, \vee, \wedge, \times)$ is not an *l*-semigroup since P_4 does not hold for multiplication. However, $(\mathbb{R}^+, \vee, \wedge, \times)$ is an *l*-group.

An *sl*-semigroup $\mathbb{F}$ has an *identity element* under $\times$ if there exists an element ϕ in $\mathbb{F}$ such that $\phi \times a = a \times \phi = a$ for all $a \in \mathbb{F}$. It has a *null element* η if $\eta \vee a = a$ and $\eta \times a = a \times \eta = \eta$ for all $a \in \mathbb{F}$. If the *sl*-semigroup $(\mathbb{F}, \vee, \times)$ is also an *sl*-semigroup $(\mathbb{F}, \wedge, \times')$ under a semilattice operation $\wedge$ and a semigroup operation $\times'$ and satisfies

$$a \vee (b \wedge a) = a \wedge (b \vee a) = a \;\; \forall \;\; a, b \in \mathbb{F},$$

then we say that $\mathbb{F}$ is an *sl-semigroup with duality*. If the operations $\times$ and $\times'$ coincide, then the operation $\times$ is called *self-dual*. An *sl*-semigroup with duality and self-dual multiplication is simply a lattice-ordered semigroup. The *l*-group $(\mathbb{R}, \vee, \wedge, +)$ has the identity element 0 but no null element, as there is no "smallest" element in $\mathbb{R}$. As another example, the *l*-semigroup $(\mathbb{R}_{-\infty}, \vee, \wedge, +)$, where $+$ denotes the extended real addition $a + (-\infty) = -\infty + a = -\infty \;\; \forall a \in \mathbb{R}_{-\infty}$, has the null element $-\infty$ but has no identity element, as $-\infty$ has no inverse under extended real addition.

It has been shown (Birkhoff [3]) that except in the trivial case where $\mathbb{F} = \{0\}$ a partially ordered group cannot have universal bounds. Thus an *l*-group cannot be a *complete* lattice (unless it is $\{0\}$), that is, each subset U of $\mathbb{F}$ cannot have a lub and a glb in $\mathbb{F}$. In particular, the *l*-group $(\mathbb{R}, \vee, \wedge, +)$, cannot be a complete lattice. Our particular interests focus on the extension of an *l*-group in a well-defined manner to include the universal bounds $-\infty$ and ∞, and hence the resulting structure will not be as strong as an *l*-group. Note that if we adjoin the element $-\infty$ (or $+\infty$) to the *l*-group $\mathbb{F}$, the structure $(\mathbb{F}_{-\infty}, \vee, \wedge, \times)$ (or $(\mathbb{F}_{\infty}, \vee, \wedge, \times)$) degenerates to an *l*-semigroup since the element $-\infty$ (or $+\infty$) cannot have a inverse under $\times$. Here, $\mathbb{F}_{-\infty} = \mathbb{F} \cup \{-\infty\}$ and $\mathbb{F}_{\infty} = \mathbb{F} \cup \{+\infty\}$. This does not turn out to be as much of a disadvantage as one might think, as an arbitrary *l*-group $\mathbb{F}$ can be extended to include the elements $-\infty$ and $+\infty$ in a well-defined manner under both operations $\vee$ and $\times$ in the following way. Let $(\mathbb{F}, \vee, \wedge, \times)$ be an *l*-group, and let $\mathbb{F}_{\pm\infty} = \mathbb{F} \cup \{+\infty, \infty\}$, where $-\infty < a < \infty \;\; \forall \;\; a \in \mathbb{F}$. The group operation $\times$ is extended in the following manner. If $a, b \in \mathbb{F}$, then $a \times b$ is already defined. Let $\times' = \times$ be the self-dual multiplication on elements of $\mathbb{F}$, that is,

$$a \times' b = a \times b \;\; \forall \; a, b \in \mathbb{F}.$$

Otherwise, we have

$$\begin{array}{ll}
a \times -\infty = -\infty \times a = -\infty & a \in \mathbb{F}_{-\infty} \\
a \times \infty = \infty \times a = \infty & a \in \mathbb{F}_{\infty} \\
a \times' -\infty = -\infty \times' a = -\infty & a \in \mathbb{F}_{-\infty} \\
a \times' \infty = \infty \times' a = \infty & a \in \mathbb{F}_{\infty} \\
(-\infty) \times \infty = \infty \times (-\infty) = -\infty & \\
(-\infty) \times' \infty = \infty \times' (-\infty) = \infty &
\end{array}$$

Hence, the element $-\infty$ acts as a null element in the system $(\mathbb{F}_{\pm\infty}, \vee, \times)$ and the element $+\infty$ acts as a null element in the system $(\mathbb{F}_{\pm\infty}, \wedge, \times')$. However, as shown by the last two equations, the multiplications $\times$ and $\times'$ introduce an asymmetry between $-\infty$ and $+\infty$. The resultant structure $(\mathbb{F}_{\pm\infty}, \vee, \wedge, \times, \times')$ is called a *bounded l*-group, and, in fact, is a distributive lattice. Note that this extension is valid for any arbitrary *l*-group. Two familiar examples of bounded *l*-groups are $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$ and $\left(\mathbb{R}^{+}_{\pm\infty}, \vee, \wedge, \times, \times'\right)$. Here, $\mathbb{R}^{+}_{\pm\infty} = \mathbb{R}^{+} \cup \{-\infty, \infty\} = \{r \in \mathbb{R} : r > 0\} \cup \{-\infty, \infty\}$. Note that $(\mathbb{R}, \vee, \wedge, +)$ is isomorphic to $(\mathbb{R}^{+}, \vee, \wedge, \times)$ both as a group and as a lattice, and hence their extensions to *l*-groups will be isomorphic as well. These extensions play a key role in lattice spaces (Section **3.12**) and have important consequences in the decomposition of templates and structuring elements (Chapter **7**).

We now return to our original question concerning rings that are partially ordered.

3.11.12 **Definition.** A *partially ordered ring* or *po-ring* is a ring R which is also a partially ordered set under a relation $\succeq$ which satisfies properties P_1 and P_2. A *lattice ordered ring* or *l-ring* is a *po*-ring R which is a lattice defined by $\succeq$.

3.11.13 **Example:** Since $(\mathbb{R}, \vee, \wedge, +, \times)$ satisfies P_1 and P_2, $\mathbb{R}$ is an *l*-ring.

As another example consider $M_{n\times n}(\mathbb{F})$, the set of all $n \times n$ matrices with entries from $\mathbb{F}$. If $\mathbb{F}$ is a totally ordered field with order $\succeq$, then by defining $\mathbf{a} \succeq \mathbf{b} \Leftrightarrow a_{ij} \succeq b_{ij} \;\; \forall i,j = 1,2,\ldots,n$, with $\mathbf{a}, \mathbf{b} \in M_{n\times n}(\mathbb{F})$, it is easy to see that $M_{n\times n}(\mathbb{F})$ satisfies P_1 and P_2.

If R is a *po*-ring, then R is a *po*-group with respect to its additive operation. Suppose $x \succeq 0$ and $y \succeq z$. Since R is a *po*-group, $x \succeq 0$ and $y \succeq z$ is true $\Leftrightarrow$ $x \succeq 0$ and $y - z \succeq 0$. Applying P_2 to the latter statement, we obtain $x(y-z) \succeq 0$ and, hence, $xy - xz \succeq 0$ or equivalently $xy \succeq xz$. Conversely, if $xy \succeq xz$, then $xy - xz \succeq 0$ and hence $x(y-z) \succeq 0$. Thus, *if* our original hypothesis $x \succeq 0$ and $y \succeq z$ implies that $xy \succeq xz$, *then* it follows that $x \succeq 0$ and $y - z \succeq 0$ implies $x(y-z) \succeq 0$. We have just proven the following:

3.11.14 **Theorem.** *In any po-ring property P_2 is equivalent to property*

$$P_5 : \qquad x \succeq 0 \;\text{ and }\; y \succeq z \quad \Rightarrow \quad xy \succeq xz$$

Since $x \vee y \succeq x, y$ and $x, y \succeq x \wedge y$, Theorem **3.11.14** has the following consequence:

3.11.15 **Corollary.** *In any l-ring*

$$P_6 : \qquad z \succeq 0 \quad \Rightarrow \quad z(x \vee y) \succeq zx \vee zy \;\text{ and }\; zx \wedge zy \succeq z(x \wedge y).$$

This is the best we can achieve for general *l*-rings. However, in totally ordered rings the inequalities in P_6 can be replaced by equalities.

3.11.16 **Corollary.** *Property P_3 holds in any totally ordered ring R.*

Proof: Since R is totally ordered, either $x \vee y = x$ or $x \vee y = y$. If $x \vee y = x$, then $zx = z(x \vee y) \succeq zx \vee zy$ and, therefore, $z(x \vee y) = zx \vee zy$. The case $x \vee y = y$ and the remainder of the proof are just as simple.

Q.E.D.

In order for P_3 to hold, total order, although sufficient, is not a necessary condition; the *l*-ring $(\mathbb{R}^{\mathbf{X}}, \vee, \wedge, +, \times)$ satisfies P_3 but is not totally ordered.

3.12 Minimax Algebra

In recent years lattice based matrix operations have found widespread applications in the engineering sciences. In these applications, the usual matrix operations of addition and multiplication are replaced by corresponding lattice operations. For example, given the bounded *l*-group $(\mathbb{F}_{\pm\infty}, \vee, \times)$ and $A = (a_{ij})$, $B = (b_{ij})$ two $m \times n$ matrices with entries in $\mathbb{F}_{\pm\infty}$, then the *pointwise maximum*, $A \vee B$, of A and B, is the $m \times n$ matrix C defined by

$$A \vee B = C\,, \text{ where } c_{ij} = a_{ij} \vee b_{ij}\,.$$

If A is $m \times p$ and B is $p \times n$, then the *max product* of A and B is the matrix $C = A \times B$, where

$$c_{ij} = \bigvee_{k=1}^{p} (a_{ik} \times b_{kj})\,.$$

Observe that this product is analogous to the usual matrix product

$$c_{ij} = \sum_{k=1}^{p} (a_{ik} \times b_{kj})$$

with the symbol $\sum$ replaced by $\bigvee$. Since $\bigvee$ replaces $\sum$ in our definition, the pointwise maximum can be thought of as matrix addition.

3.12.1 Example:

(i) Consider the bounded *l*-group $(\mathbb{R}_{\pm\infty}, \vee, +)$. Then

$$\begin{bmatrix} -\infty & 6 & -2 & 2 \\ 7 & -5 & 10 & -4 \\ 8 & 4 & +\infty & 9 \\ -3 & -\infty & 1 & -7 \\ -1 & 1 & 0 & 5 \end{bmatrix} \bigvee \begin{bmatrix} 2 & 7 & +\infty & -\infty \\ -\infty & 0 & 8 & 0 \\ -3 & 6 & 12 & 4 \\ -1 & -\infty & 2 & -3 \\ -2 & 1 & -1 & 6 \end{bmatrix} = \begin{bmatrix} 2 & 7 & +\infty & 2 \\ 7 & 0 & 10 & 0 \\ 8 & 6 & +\infty & 9 \\ -1 & -\infty & 2 & -3 \\ -1 & 1 & 0 & 6 \end{bmatrix},$$

which provides an illustration of the pointwise maximum of two 5×4 matrices with entries from $\mathbb{R}_{\pm\infty}$.

(ii) For an illustration of the max product we use a 5×4 and a 4×3 matrix with entries from $\mathbb{R}_{\pm\infty}$:

$$\begin{bmatrix} -\infty & 6 & -2 & 2 \\ 7 & -5 & 10 & -4 \\ 8 & 4 & 11 & 9 \\ -3 & +\infty & 1 & -7 \\ -1 & 1 & 0 & 5 \end{bmatrix} \times \begin{bmatrix} -\infty & 6 & -2 \\ 7 & -5 & 10 \\ 8 & 4 & 11 \\ -1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 13 & 3 & 16 \\ 18 & 14 & 21 \\ 19 & 15 & 22 \\ +\infty & +\infty & +\infty \\ 8 & 6 & 11 \end{bmatrix}.$$

Here

$$c_{ij} = \bigvee_{k=1}^{p} (a_{ik} + b_{kj}) \, .$$

If $(\mathbb{F}_{\pm\infty}, \wedge, \times')$ is the dual of $(\mathbb{F}_{\pm\infty}, \vee, \times)$, then the *dual* or *min product* of A and B is the matrix $C = A \times' B$, where

$$c_{ij} = \bigwedge_{k=1}^{p} \left(a_{ik} \times' b_{kj}\right) \, .$$

Similarly, the *pointwise minimum* $A \wedge B$ of two matrices of the same size is defined as

$$A \wedge B = C \, , \text{ where } c_{ij} = a_{ij} \wedge b_{ij} \, .$$

Lattice induced matrix operations lead to an entirely different perspective of a class of nonlinear transformations. These ideas were applied by Shimbel [21] to communications networks, and to machine scheduling by Cuninghame-Green [7, 8] and Giffler [13]. Others have discussed their usefulness in applications to shortest path problems in graphs [18, 2, 6, 1]. Additional examples are given in [9], primarily in the field of operations research. Another useful application to image processing was developed by Ritter and Davidson, [19] and [10].

While lattice theory and lattice-ordered groups have only marginal connections to the computational aspects of linear algebra, Cuninghame-Green developed a novel nonlinear matrix calculus based on the min and max product, called *minimax algebra*, which is very reminiscent of linear algebra [9]. Problems notated using the *minimax products* take on the flavor of problems in linear algebra. By allowing for the minimax matrix products to take on the character of the familiar matrix products, concepts analogous to those in linear algebra such as solutions to systems of equations, linear dependence and independence, rank, seminorms, eigenvalues and eigenvectors, spectral inequalities, and invertible and equivalent matrices, can be formulated. Originally, many of these concepts were developed primarily to help solve operations research types of problems. Our interest in these notions is due to their applicability to image processing problems.

An *sl*-semigroup $(\mathbb{F}, \vee, \times)$ satisfying the axioms

B_1 $\quad x \times (y \vee z) = (x \times y) \vee (x \times z)$

B_2 $\quad (y \vee z) \times x = (y \times x) \vee (z \times x)$

$\forall\, x, y, z \in \mathbb{F}$ is called a *belt*. Viewing $\times$ as multiplication and $\vee$ as addition provides $\mathbb{F}$ with a ring like appearance, hence the name "belt." If, in addition, $\mathbb{F}$ is a lattice with another semigroup operation $\times'$ satisfying

B_1' $\quad x \times' (y \wedge z) = (x \times' y) \wedge (x \times' z)$

B_2' $\quad (y \wedge z) \times' x = (y \times' x) \wedge (z \times' x),$

then $\mathbb{F}$ is called a *belt with duality.*

If the multiplication $\times$ and the dual multiplication $\times'$ coincide, then we call the multiplication *self-dual.* Obviously, any *l*-semigroup is a belt with self-duality. If the *l*-semigroup is actually an *l*-group, then each element $x \in \mathbb{F}$ has a unique *multiplicative* inverse x^{-1}; by analogy with division rings, we call such a belt a *division belt*.

Let $(\mathbb{F}_1,\vee)$ and $(\mathbb{F}_2,\vee)$ be two semilattices. A function $f : \mathbb{F}_1 \to \mathbb{F}_2$ is called a *semilattice homomorphism* if

$$f(a \vee b) = f(a) \vee f(b)$$

for all a, b in $\mathbb{F}_1$. If $(\mathbb{F}_1,\vee,\times)$ and $(\mathbb{F}_2,\vee,\times)$ are two belts and $f : \mathbb{F}_1 \to \mathbb{F}_2$ is a semilattice homomorphism that satisfies

$$f(a \times b) = f(a) \times f(b), \ \ \forall \ \ a, b \in \mathbb{F},$$

then f is called a *belt* homomorphism. A belt (or semilattice) *isomorphism* or *automorphism* is given the usual meaning.

3.12.4 Example:

Define $f : \mathbb{R} \to \mathbb{R}^+$ by $f(x) = e^x$. Then $f(x \vee y) = f(x) \vee f(y)$ and $f(x + y) = f(x) \times f(y)$. It is trivial to show that f is a belt isomorphism. Furthermore, $\mathbb{R}$ and $\mathbb{R}^+$ are *commutative* belts, that is, the multiplication $\times$ commutes.

Let $(\mathbb{F},\vee,\wedge,\times)$ be an *l*-group with identity ϕ and let $(\mathbb{T},\vee)$ be a semilattice. Suppose we have a right multiplication of elements of $\mathbb{T}$ by elements of $\mathbb{F}$:

$$x \times \lambda \in \mathbb{T} \quad \forall\, x \in \mathbb{T}, \ \text{and}\, \forall\, \lambda \in \mathbb{F}.$$

We call $(\mathbb{T},\vee)$ a (*right*) *semilattice space* over $\mathbb{F}$ if the following four conditions are satisfied for all $x, y \in \mathbb{T}$ and for all $\lambda, \mu \in \mathbb{F}$:

$$\begin{aligned}(x \times \lambda) \times \mu &= x \times (\lambda \times \mu)\\ (x \vee y) \times \lambda &= (x \times \lambda) \vee (y \times \lambda)\\ x \times (\lambda \vee \mu) &= (x \times \lambda) \vee (x \times \mu)\\ x \times \phi &= x\ .\end{aligned}$$

A right semilattice space is also called a *right s-lattice space* or simply a *space over* $\mathbb{F}$. If $\mathbb{T}$ and $\mathbb{F}$ are known, then shall simply say that $\mathbb{T}$ is a *space.* A *subspace* is a subset of a space which itself is a space over $\mathbb{F}$.

Semilattice spaces play the role of vector spaces in the minimax theory. For example, if $\mathbb{T} = \mathbb{R}^n$ and $\mathbb{F} = \mathbb{R}$, and we define

$$\mathbf{a} \times r = \mathbf{b}, \ \ \text{where}\ b_i = a_i + r,\ i = 1, \ldots, n$$

then it is easily verified that $\times$ satisfies the above four axioms. Here, $(\mathbb{R}^{\mathbf{n}},\vee)$ is a semilattice under coordinate-wise maximum. Thus we can regard $\mathbb{R}^n$ as a right semilattice space over $\mathbb{R}$. Similarly, we can define a *left semilattice space* over $\mathbb{F}$, using the left versions of the above four conditions. A *two-sided space* is a triple $(\mathbb{L},\mathbb{T},\mathbb{F})$ satisfying the following three axioms:

S_1 : $\mathbb{L}$ is a belt and $\mathbb{T}$ is a left space over $\mathbb{L}$

S_2 : $\mathbb{L}$ is a belt and $\mathbb{T}$ is a right space over $\mathbb{L}$

S_3 : $\alpha \times (x \times \beta) = (\alpha \times x) \times \beta \quad \forall \alpha \in \mathbb{L},\ \forall x \in \mathbb{T},$ and $\forall \beta \in \mathbb{F}.$

Let $(\mathbb{F},\vee,\times)$ be a belt. An important class of spaces over $\mathbb{F}$ is the class of function spaces. Here the *s*-lattice $(\mathbb{T},\vee)$ is replaced by the lattice $\left(\mathbb{F}^{\mathbf{X}},\vee\right)$. Such spaces are naturally two-sided. We shall only be interested in the case were $card(\mathbf{X}) = n < \infty$. Thus we can view $\left(\mathbb{F}^{\mathbf{X}},\vee\right)$ as the space of *n*-tuples $(\mathbb{F}^n,\vee)$.

When discussing conjugacy in linear operator theory, two approaches are commonly used. One defines the conjugate of a given space $\mathbf{S}$ as a special set $\mathbf{S}^*$ of linear scalar-valued functions defined on $\mathbf{S}$. The other involves defining an *involution* (a one-to-one function) taking $x \in \mathbf{S}$ to $x^* \in \mathbf{S}^*$ that satisfies certain properties. The situation is slightly more complicated in the case of lattice transforms.

Let $(\mathbb{F},\vee,\times)$ and $(\mathbb{T},\wedge,\times')$ be two belts. We say that $(\mathbb{T},\wedge,\times')$ is *conjugate* to $(\mathbb{F},\vee,\times)$ if there is a one-to-one correspondence $g : \mathbb{F} \to \mathbb{T}$ satisfying the following two conditions:

$$\forall x,y \in \mathbb{F},\ \ g(x \vee y) = g(x) \wedge g(y)$$
$$\forall x,y \in \mathbb{F},\ \ g(x \times y) = g(x) \times' g(y)\,.$$

In lattice theory, g is called a *dual isomorphism*. Note that conjugacy is a symmetric relation. If $(\mathbb{F},\vee,\wedge)$ is an *s*-lattice with duality and $g : \mathbb{F} \to \mathbb{F}$ satisfying

$$\forall x,y \in \mathbb{F},\ \ g(x \vee y) = g(x) \wedge g(y)\,,$$

then we say that $\mathbb{F}$ is *self-conjugate*. If $(\mathbb{F},\vee,\times,\wedge,\times')$ is a belt with duality, we say that $\mathbb{F}$ is *self-conjugate* whenever $(\mathbb{F},\wedge,\times')$ is conjugate to $(\mathbb{F},\vee,\times)$.

In particular, every division belt is self-conjugate under the one-to-one correspondence $x^* = x^{-1}$, and every bounded *l*-group is self-conjugate under the one-to-one correspondence $(-\infty)^* = \infty,\ \infty^* = -\infty$, and $x^* = x^{-1}$ if x is finite. Thus, if $\mathbb{F}_{\pm\infty}$ is a *multiplicative* bounded *l*-group and $r \in \mathbb{F}_{\pm\infty}$, then the *additive conjugate* of r is the unique element r^* defined by

$$r^* \equiv \begin{cases} r^{-1} & \text{if } r \in \mathbb{F} \\ -\infty & \text{if } r = +\infty \\ +\infty & \text{if } r = -\infty \end{cases}. \tag{3.12.1}$$

Here, r^{-1} is the inverse of r under the group operation $\times$. Therefore, $(r^*)^* = r$. This gives the following relation for all r,u in $\mathbb{F}_{\pm\infty}$:

$$r \wedge u = (r^* \vee u^*)^*.$$

If $(\mathbb{F},\vee,\times,\wedge,\times')$ is a belt with duality, then we say that the space $(\mathbb{T},\vee)$ over $\mathbb{F}$ has a *duality* if a dual addition $\wedge$ is defined such that $(\mathbb{T},\vee,\wedge)$ is an s-lattice with duality and $(\mathbb{T},\wedge)$ is a space over $(\mathbb{F},\wedge,\times')$.

We now return to the nonlinear matrix algebra induced by belts. Let $M_{n\times p}(\mathbb{F})$ denote the set of all $n \times p$ matrices with values in the belt $\mathbb{F}$. The following are some basic definitions and properties:

(1) Scalar multiplication of a matrix A by an element $\lambda \in \mathbb{F}$ is defined by

$$(a_{ij}) \times \lambda = (a_{ij} \times \lambda)$$
$$\lambda \times (a_{ij}) = (\lambda \times a_{ij})$$

for all $A = (a_{ij}) \in M_{n\times p}(\mathbb{F})\,,\ \ \lambda \in \mathbb{F}$;

(2) $(M_{n\times p}(\mathbb{F}), \vee)$ is an *s*-lattice and a function space over $(\mathbb{F}, \vee, \times)$;

(3) $(M_{n\times n}(\mathbb{F}), \vee, \times)$ is a belt;

(4) $(M_{n\times p}(\mathbb{F}), \vee)$ is a left space over the belt $(M_{n\times n}(\mathbb{F}), \vee, \times)$;

(5) For all $A \in M_{m\times p}(\mathbb{F})$, $B, C \in M_{p\times n}(\mathbb{F})$, $\lambda \in \mathbb{F}$,

$$A \times (B \vee C) = (A \times B) \vee (A \times C)$$
$$A \times (B \times \lambda) = (A \times B) \times \lambda\,.$$

The proofs of these properties are straight forward and can be found in [9].

Since the *s*-lattice $(M_{n\times 1}(\mathbb{F}), \vee)$ is isomorphic to the s-lattice $\mathbb{F}^n$, we have that $\mathbb{F}^n$ is a function space over $\mathbb{F}$ as well as a space over $M_{n\times n}(\mathbb{F})$. This mimics the classical role of matrices as linear transformation of spaces of n-tuples.

As in real or complex valued matrix theory, two matrices of prime importance in minimax theory are the identity matrix and the null matrix. Suppose that the belt $\mathbb{F}$ has identity and null elements $\phi \text{ and } -\infty$, respectively. We define the *identity matrix* $I \in M_{n\times n}(\mathbb{F})$ by

$$I = \begin{pmatrix} \phi & -\infty & \cdot & \cdot & -\infty \\ -\infty & \phi & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & -\infty \\ -\infty & \cdot & \cdot & -\infty & \phi \end{pmatrix}$$

and the *null matrix* $\Phi \in M_{n\times n}(\mathbb{F})$ by

$$\Phi = \begin{pmatrix} -\infty & \cdot & \cdot & \cdot & -\infty \\ \cdot & -\infty & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ -\infty & \cdot & \cdot & \cdot & -\infty \end{pmatrix}.$$

Thus we have $\forall A \in M_{n\times n}(\mathbb{F})$,

$$I \times A = A \times I = A$$
$$A \vee \Phi = \Phi \vee A = A$$
$$A \times \Phi = \Phi \times A = \Phi.$$

In the bounded *l*-group $\mathbb{R}_{\pm\infty}$ we have

$$I = \begin{pmatrix} 0 & -\infty & \cdot & \cdot & -\infty \\ -\infty & 0 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & -\infty \\ -\infty & \cdot & \cdot & -\infty & 0 \end{pmatrix}$$

and in $\mathbb{R}^{+}_{\pm\infty}$ we have

$$I = \begin{pmatrix} 1 & -\infty & \cdot & \cdot & -\infty \\ -\infty & 1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & -\infty \\ -\infty & \cdot & \cdot & -\infty & 1 \end{pmatrix}.$$

Conjugacy extends to matrices if the underlying value set is itself a self-conjugate belt. This is stated in the next theorem.

3.12.7 Theorem. (Cunninghame-Greene [9]) *If* $(\mathbb{F},\vee,\times,\wedge,\times')$ *is a self-conjugate belt, then* $(M_{n\times n}(\mathbb{F}),\vee,\times,\wedge,\times')$ *is a self-conjugate belt.*

Here the conjugate of a matrix $A = (a_{ij})$ with entries in $\mathbb{F}$ or $\mathbb{F}_{\pm\infty}$ is the matrix $A^* = (b_{ij})$, where $b_{ij} = [a_{ji}]^*$ and $[a_{ji}]^*$ is the additive conjugate of a_{ji} defined earlier. The notions of pointwise minimum and dual product could have been defined in terms of conjugation since

$$A \wedge B = (A^* \vee B^*)^*$$

and

$$A \times' B = (B^* \times A^*)^*$$

for appropriately sized matrices.

While semilattice spaces over *l*-groups play the role of vector spaces in minimax theory, *linear* homomorphisms of semilattice spaces take on the role of linear transformations.

3.12.8 Definition. Let $(\mathbb{L},\vee)$ and $(\mathbb{T},\vee)$ be given spaces over a belt $(\mathbb{F},\vee,\times)$. An s-lattice homomorphism $g : (\mathbb{L},\vee) \to (\mathbb{T},\vee)$ is called *right linear* (*over* $\mathbb{F}$) if

$$g(x \times \lambda) = g(x) \times \lambda \ \ \forall x \in \mathbb{L},\ \forall \lambda \in \mathbb{F}.$$

We denote the set of all right linear homomorphisms from $\mathbb{L}$ to $\mathbb{T}$ over $\mathbb{F}$ by $Hom_{\mathbb{F}}(\mathbb{L},\, \mathbb{T})$.

In linear algebra, we characterize linear transformations of vector spaces entirely in terms of matrices. A natural question to ask is whether or not a similar classification holds in minimax algebra. The following results give necessary and sufficient conditions for this to be the case.

3.12.9 Theorem. (Cunninghame-Greene [9]) *If* $\mathbb{F}$ *is a belt which has an identity element* ϕ *and a null element* $-\infty$*, then for all integers* $m,\, n \geq 1,\ M_{m\times n}(\mathbb{F})$ *is isomorphic to* $Hom_{\mathbb{F}}(\mathbb{F}^n,\, \mathbb{F}^m)$.

The belt operations of $\vee$ and $\times$ on $Hom_{\mathbb{F}}(\mathbb{F}^n,\, \mathbb{F}^m)$ are the naturally induced operations from the belt $\mathbb{F}^m$ defined by $(f \vee g)(x) = f(x) \vee g(x)$ and $(f \times g)(x) = f(x) \times g(x)\ \forall f,\, g \in Hom_{\mathbb{F}}(\mathbb{F}^n,\, \mathbb{F}^m)$.

3.12.10 Corollary: *Suppose $n \in \mathbb{Z}$ with $n > 1$. If $\mathbb{F}$ is a belt, then a necessary and sufficient condition that $M_{m\times n}(\mathbb{F})$ is isomorphic to $Hom_{\mathbb{F}}(\mathbb{F}^n, \mathbb{F}^m)$ for all integers $m \geq 1$ is that $\mathbb{F}$ has an identity element ϕ with respect to $\times$ and a null element $-\infty$ with respect to $\vee$.*

We call a matrix $A \in M_{m\times n}(\mathbb{F})$ a *lattice transform.* Since $Hom_{\mathbb{F}}(\mathbb{F}^n, \mathbb{F}^m)$ is isomorphic to $M_{m\times n}(\mathbb{F})$, we shall usually refer to $f \in Hom_{\mathbb{F}}(\mathbb{F}^n, \mathbb{F}^m)$ as a lattice transform instead of a right linear homomorphism. In particular, lattice transforms are to minimax algebra what linear transforms are to linear algebra.

Much of what has been established in this section can be expressed in the context of dual lattice-ordered semigroups. However, we wish to study these structures from a different perspective. The extension of the belt operations to matrices allows us to view matrices as operators on spaces of *n*-tuples in a way similar to vector-space transformations even though these operators are non-linear due to the lattice structure of the underlying set $\mathbb{F}$. This point of view will be especially useful when one is interested in optimizing non-linear image transforms that correspond to lattice transforms (Chapter **7**).

The minimax algebra that we will be mostly concerned with is based on the two *isomorphic* bounded *l*-groups $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$ and $\left(\mathbb{R}^{+}_{\pm\infty}, \vee, \wedge, \times, \times'\right)$. The substructure $(\{-\infty, 0, \infty\}, \vee, \wedge, +, +')$ of $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$, which is a *3–element bounded l-group*, is of particular interest in Boolean image processing. The sublattice $(\{-\infty, 0\}, \vee, \wedge)$ of the 3–element bounded *l*-group is a Boolean algebra with complementation $(-\infty)' = 0$ and $0' = -\infty$. By setting $\texttt{OR} = \vee$, $\texttt{AND} = \wedge$, $\texttt{FALSE} = -\infty$, and $\texttt{TRUE} = 0$, we obtain the Boolean algebra commonly used in computer science.

3.13 Heterogeneous Algebras

In the previous sections we considered several algebraic systems, or *algebras*, such as groups, rings, linear algebras, lattice algebras, and minimax algebras. These algebras are all special cases of a more general concept known as a *heterogeneous algebra*. Image algebra is a particularly applicable example of a heterogeneous algebra.

Our basic definition of a heterogeneous algebra is due to Birkhoff and Lipson [4].

3.13.1 Definition. An *algebra* $\mathcal{A}$ is a pair $\mathcal{A} = (\mathcal{F}, \mathcal{O})$, where

(1) $\mathcal{F} = \{\mathbb{F}_\lambda\}_{\lambda\in\Lambda}$ is a family of non-empty sets of different types of elements and the subscripts λ are members of some indexing set Λ, and

(2) $\mathcal{O} = \{\bigcirc_\kappa\}_{\kappa\in\mathrm{K}}$ is a set of finitary operations (for some indexing set K), where each $\bigcirc_\kappa \in \mathcal{O}$ is a mapping of the Cartesian product of some of the $\mathbb{F}_\lambda$'s to another; that is,

$$\bigcirc_\kappa : \prod_{i=1}^{n_\kappa} \mathbb{F}_{\lambda_{i(\kappa)}} \to \mathbb{F}_{\lambda(\kappa)},$$

and each $\mathbb{F}_{\lambda_{i(\kappa)}}, \mathbb{F}_{\lambda(\kappa)} \in \mathcal{F}$.

In this definition the notation n_κ, $\lambda(\kappa)$, and $i(\kappa)$ means that the indexing depends on the operation $\bigcirc_\kappa$. In particular,

$$\bigcirc_\kappa : \left(x_{\lambda_{1(\kappa)}}, \ldots, x_{\lambda_{n_k(\kappa)}}\right) \to x_{\lambda(\kappa)} \in \mathbb{F}_{\lambda(\kappa)}.$$

The operation $\bigcirc_\kappa$ is *unary* if $n_\kappa = 1$, *binary* if $n_\kappa = 2$, *ternary* if $n_\kappa = 3$, etc. The elements $\mathbb{F}_\lambda$ of $\mathcal{F}$ are called the *sets of operands* of $\mathcal{A}$, and the elements $\bigcirc_\kappa \in \mathcal{O}$ are called the *operators* of (or *operations* on) $\mathcal{A}$.

An algebra $\mathcal{A}$ is called a *homogeneous* or *single valued* algebra if $\mathcal{F}$ contains only one element — i.e., $\mathcal{F} = \{\mathbb{F}\}$, otherwise $\mathcal{A}$ is called a *heterogeneous* or *many valued* algebra. If $\mathcal{A}$ is single valued, we simply write $(\mathbb{F}, \mathcal{O})$ instead of $(\{\mathbb{F}\}, \mathcal{O})$ and if the operations are tacitly understood, we write $\mathbb{F}$ in place of $(\{\mathbb{F}\}, \mathcal{O})$.

If $\mathcal{A} = (\mathcal{F}, \mathcal{O})$ and $\mathcal{B} = (\mathcal{G}, \mathcal{U})$ are two algebras with $\mathcal{F} = \{\mathbb{F}_\lambda\}$ and $\mathcal{G} = \{\mathbb{G}_\mu\}$, then $\mathcal{B}$ is called a *subalgebra* of $\mathcal{A}$ if and only if $\mathcal{U} \subset \mathcal{O}$ and for each $\mathbb{G}_\mu \in \mathcal{G}$ there exists a $\mathbb{F}_\lambda \in \mathcal{F}$ such that $\mathbb{G}_\mu \subset \mathbb{F}_\lambda$.

3.13.2 Examples:

(i) A linear algebra $\mathcal{L}(\mathbb{F}) = (\mathcal{F}, \mathcal{O})$, where $\mathcal{F} = \{\mathcal{L}, \mathbb{F}\}$ consists of a set $\mathcal{L}$ of vectors and a set $\mathbb{F}$ of scalars, and $\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{O}_3$ (see Section **3.9**). The scalars form a field under the operations $\mathcal{O}_1 = \{+, \times\}$ and the set $\mathcal{L}$ forms a ring under the operations $\mathcal{O}_2 = \{\dot{+}, *\}$. In addition, there is a fifth operation $\mathbb{F} \times \mathcal{L} \to \mathcal{L}$ of scalar multiplication, $\mathcal{O}_3 = \{\cdot\}$, satisfying the vector space axioms V_1 through V_5 (**3.7**) and the linear algebra axiom L_5. Since $card(\mathcal{F}) > 1$, $\mathcal{L}(\mathbb{F})$ is a heterogeneous algebra. Note that the ring $\{\mathcal{L}, \dot{+}, *\}$ is a homogeneous subalgebra of $\mathcal{L}(\mathbb{F})$.

(ii) A finite state machine $(\{S, I, O\}, \{f, g\})$ is a heterogeneous algebra. Here
S = set of states,
I = set of input symbols,
O = set of output symbols,
$f : S \times I \to S$ is the state transition function, and
$g : S \to O$ is the state output function.

If $\mathcal{A} = (\{\mathbb{F}_\lambda\}, \mathcal{O})$ and $\mathcal{B}_j = (\{\mathbb{G}_{j,\lambda}\}, \mathcal{O})$ is a family of subalgebras with $\mathbb{G}_{j,\lambda} \subset \mathbb{F}_\lambda$ for every index λ, then the intersection $\bigcap_j \mathcal{B}_j = (\{\cap_j \mathbb{G}_{j,\lambda}\}, \mathcal{O})$. Higgins proved the following theorem [14].

3.13.3 Theorem. *Any intersection of subalgebras of $\mathcal{A}$ is a subalgebra of $\mathcal{A}$.*

Another consequence of the definition of subalgebras of an algebra is that the subalgebras of an algebra form a complete lattice [3].

The notion of a homomorphism between algebraic systems as defined in the previous sections has its natural generalization to heterogeneous algebras.

3.13.4 Definition. Let $\mathcal{A} = (\{\mathbb{F}_\lambda\}, \mathcal{O})$ and $\mathcal{B} = (\{\mathbb{G}_\lambda\}, \mathcal{O}')$ be two algebras. A *homomorphism* $\psi : \mathcal{A} \to \mathcal{B}$ is a set of functions $\psi_\lambda : \mathbb{F}_\lambda \to \mathbb{G}_\lambda$ (one for each $\mathbb{F}_\lambda$) such that for any operation

$$\bigcirc_\kappa : \prod_{i=1}^{n_\kappa} \mathbb{F}_{\lambda_{i(\kappa)}} \to \mathbb{F}_{\lambda(\kappa)}$$

in $\mathcal{O}$ there exists an $\bigcirc'_\kappa \in \mathcal{O}'$, with

$$\bigcirc'_\kappa : \prod_{i=1}^{n_\kappa} \mathbb{G}_{\lambda_{i(\kappa)}} \to \mathbb{G}_{\lambda(\kappa)} \, ,$$

such that the composition

$$\psi_{\lambda(\kappa)} \circ \bigcirc_\kappa = \bigcirc'_\kappa \circ \left(\psi_{\lambda_{i(1)}}, \psi_{\lambda_{i(2)}}, \ldots, \psi_{\lambda_{i(n_\kappa)}}\right)$$

holds.

As usual, homomorphisms that are one-to-one and onto — meaning $\psi_\lambda : \mathbb{F}_\lambda \to \mathbb{G}_\lambda$ is one-to-one and onto for each λ — are called *isomorphisms.* For instance, if X is a finite set with $card(X) = n$, then $\left(\mathbb{R}^X, \mathbb{R}, \mathcal{O}_1\right)$ and $(\mathbb{R}^n, \mathbb{R}, \mathcal{O}_2)$ are isomorphic. Here $\mathcal{O}_1$ and $\mathcal{O}_2$ denote the set of vector space operations for $\mathbb{R}^X$ and $\mathbb{R}^n$, respectively. The isomorphism $\psi : \left(\mathbb{R}^X, \mathbb{R}, \mathcal{O}_1\right) \to (\mathbb{R}^n, \mathbb{R}, \mathcal{O}_2)$ consists of the map $\nu : \mathbb{R}^X \to \mathbb{R}^n$ given in Example **2.8.3** and the identity map $1_\mathbb{R} : \mathbb{R} \to \mathbb{R}$.

3.14 Generalized Matrix Products

The common matrix product, the tensor or Kronecker product, the outer and inner vector product, and the minimax matrix product discussed in Section **3.12**, are all special cases of a more general matrix product associated with heterogeneous algebras.

Following the ideas of Section **3.12** we note that a semigroup $(\mathbb{F}, \gamma)$ induces, for any pair of positive integers m and n, a semigroup of matrices $(M_{n\times m}(\mathbb{F}), \gamma)$. The induced operation on the matrix semigroup is defined componentwise. More precisely, if $A = (a_{ij})_{n\times m}$ and $B = (b_{ij})_{n\times m}$ are elements of $M_{n\times m}(\mathbb{F})$, then

$$\begin{gathered} A\gamma B = E = (e_{ij})_{n\times m}\,, \ \text{where} \\ e_{ij} = a_{ij}\gamma b_{ij}\,,\ \forall\, i = 1,2,\ldots,n,\ j = 1,2,\ldots,m\,. \end{gathered} \tag{3.14.1}$$

.

We can take this concept a step further. If $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$ is any binary operation, then $\bigcirc$ induces a binary operation

$$\bigcirc : M_{n\times m}(\mathbb{E}) \times M_{n\times m}(\mathbb{G}) \to M_{n\times m}(\mathbb{F}) \tag{3.14.2}$$

defined by

$$\begin{gathered} A \bigcirc B = E = (e_{ij})_{n\times m}\,, \ \text{where} \\ e_{ij} = a_{ij} \bigcirc b_{ij}\,,\ \forall\, i = 1,2,\ldots,n,\ j = 1,2,\ldots,m\,. \end{gathered} \tag{3.14.3}$$

The componentwise product defined by Eqs. 3.14.2 and 3.14.3 is called the *generalized Hadamard product*.

In the special case where $\mathbb{E} = \mathbb{G} = \mathbb{F}$, the algebraic structure $(\mathbb{F}, \bigcirc, \gamma)$ induces a matrix structure $(M_{n\times m}(\mathbb{F}), \bigcirc, \gamma)$ that behaves very much like the structure $(\mathbb{F}, \bigcirc, \gamma)$. For example, if $\bigcirc$ distributes over γ in $\mathbb{F}$, then $\bigcirc$ distributes over γ in $M_{n\times m}(\mathbb{F})$. Similar comments can be made for commutativity and associativity.

In the subsequent discussion we follow the convention of letting $\mathbb{F}^m = M_{1\times m}(\mathbb{F})$ and view $\mathbb{F}^m$ as the set of *m*-dimensional row vectors with entries from $\mathbb{F}$. Similarly, if $m = 1$, then $(\mathbb{F}^n)' = (M_{1\times n}(\mathbb{F}))' = M_{n\times 1}(\mathbb{F})$ denotes the set of *n*-dimensional column vectors with entries from $\mathbb{F}$. Our goal is to define a class of matrix products that allow us to multiply matrices from $M_{l\times m}(\mathbb{E})$ with matrices from $M_{n\times q}(\mathbb{G})$, where *m* is not necessarily equal to *n*.

Suppose $p \in \mathbb{Z}^+$ which divides both *m* and *n*. Note that such an integer *p* always exists, namely $p = 1$. We shall establish the following two one-to-one correspondences:

$$\mathbb{Z}_l^+ \times \mathbb{Z}_m^+ \;\leftrightarrow\; \mathbb{Z}_l^+ \times \mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+ \;\text{ and }\; \mathbb{Z}_p^+ \times \mathbb{Z}_{n/p}^+ \times \mathbb{Z}_q^+ \;\leftrightarrow\; \mathbb{Z}_n^+ \times \mathbb{Z}_q^+\,,$$

where the double arrow "$\leftrightarrow$" denotes the appropriate one-to-one correspondence. Observe that the *last* factor in the 3–fold Cartesian product of the first correspondence is $\mathbb{Z}_p^+$, which is also the *first* factor in the 3–fold Cartesian product of the second correspondence. This common factor is the key ingredient in the definition of the general matrix product of order *p*.

We define the first correspondence $\mathbb{Z}_l^+ \times \mathbb{Z}_m^+ \;\leftrightarrow\; \mathbb{Z}_l^+ \times \mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+$ in terms of the *row scanning map*

$$\begin{aligned} &r_p : \mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+ \to \mathbb{Z}_m^+ \\ &\text{where } \; r_p(i,k) = (i-1)p + k, \\ &1 \le k \le p \;\text{ and }\; 1 \le i \le m/p\,. \end{aligned} \tag{3.14.4}$$

Since $r_p(i,k) < r_p(i',k') \Leftrightarrow i < i'$ or $i = i'$ and $k < k'$, r_p linearizes the array $\mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+$ using the row scanning order (see also Example **3.1.5**) as shown:

$$\begin{bmatrix} \overset{1}{(1,1)} & \overset{2}{(1,2)} & \cdots & \overset{k}{(1,k)} & \cdots & \overset{p}{(1,p)} \\ \overset{p+1}{(2,1)} & \overset{p+2}{(2,2)} & \cdots & \overset{p+k}{(2,k)} & \cdots & \overset{2p}{(2,p)} \\ \vdots & \vdots & & \vdots & & \vdots \\ \overset{(i-1)p+1}{(i,1)} & (i,2) & \cdots & \overset{(i-1)p+k}{(i,k)} & \cdots & \overset{ip}{(i,p)} \\ \vdots & \vdots & & \vdots & & \vdots \\ \overset{((m/p)-1)p+1}{(m/p,1)} & (m/p,2) & \cdots & (m/p,k) & \cdots & \overset{(m/p)p=m}{(m/p,p)} \end{bmatrix}$$

It follows that the *row-scanning order* on $\mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+$ is given by

$$(i,k) \le \big(i',k'\big) \;\Leftrightarrow\; r_p(i,k) \le r_p\big(i',k'\big)$$

or, equivalently, by

$$(i,k) \le \big(i',k'\big) \Leftrightarrow i < i' \;\text{ or }\; i = i' \;\text{ and }\; k \le k'\,.$$

We define the one-to-one correspondence

$$\begin{aligned} &f_p : \mathbb{Z}_l^+ \times \mathbb{Z}_{m/p}^+ \times \mathbb{Z}_p^+ \;\to\; \mathbb{Z}_l^+ \times \mathbb{Z}_m^+ \\ &\text{by }\; f_p : (x,y,z) \;\rightarrowtail\; (x, r_p(y,z))\,. \end{aligned}$$

The one-to-one correspondence allows us to re-index the entries of a matrix $A = (a_{s,t}) \in M_{l\times m}(\mathbb{E})$ in terms of a triple index $a_{s,(i,k)}$ by using the convention

$$\begin{aligned} &a_{s,(i,k)} = a_{s,t} \;\Leftrightarrow\; r_p(i,k) = t\,, \\ &\text{where }\; 1 \le i \le m/p \;\text{ and }\; 1 \le k \le p\,. \end{aligned} \tag{3.14.5}$$

3.14.1 Example: Suppose $l = 2$, $m = 6$ and $p = 2$. Then $m/p = 3$, $1 \le k \le p = 2$, and $1 \le i \le m/p = 3$. Hence for $A = (a_{s,t}) \in M_{2\times 6}(\mathbb{E})$, we have

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \end{pmatrix}$$

$$= \begin{pmatrix} a_{1,(1,1)} & a_{1,(1,2)} & a_{1,(2,1)} & a_{1,(2,2)} & a_{1,(3,1)} & a_{1,(3,2)} \\ a_{2,(1,1)} & a_{2,(1,2)} & a_{2,(2,1)} & a_{2,(2,2)} & a_{2,(3,1)} & a_{2,(3.2)} \end{pmatrix}.$$

The factor $\mathbb{Z}_n^+$ of the Cartesian product $\mathbb{Z}_n^+ \times \mathbb{Z}_q^+$ is decomposed in a similar fashion. Here the row-scanning map is given by

$$\begin{gathered} c_p : \mathbb{Z}_p^+ \times \mathbb{Z}_{n/p}^+ \to \mathbb{Z}_n^+ \\ \text{where } c_p(k,j) = (k-1)(n/p) + j, \\ 1 \le j \le n/p, \text{ and } 1 \le k \le p\,. \end{gathered} \tag{3.14.6}$$

This allows us to re-index the entries of a matrix $B = (b_{s,t}) \in M_{n\times q}(\mathbb{G})$ in terms of a triple index $b_{(k,j),t}$ by using the convention

$$\begin{gathered} b_{(k,j),t} = b_{s,t} \;\Leftrightarrow\; c_p(k,j) = s\,, \\ \text{where } 1 \le k \le p \text{ and } 1 \le j \le n/p\,. \end{gathered} \tag{3.14.7}$$

3.14.2 Example: Suppose $n = 4$, $q = 3$ and $p = 2$. Then $n/p = 2$, $1 \le k \le p = 2$, and $1 \le j \le n/p = 2$. Hence for $B = (b_{s,t}) \in M_{n\times q}(\mathbb{G})$, we have

$$B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{pmatrix} = \begin{pmatrix} b_{(1,1),1} & b_{(1,1),2} & b_{(1,1),3} \\ b_{(1,2),1} & b_{(1,2),2} & b_{(1,2),3} \\ b_{(2,1),1} & b_{(2,1),2} & b_{(2,1),3} \\ b_{(2,2),1} & b_{(2,2),2} & b_{(2,2),3} \end{pmatrix}.$$

The induced operation $\bigcirc$ in Eq. 3.14.2 is a componentwise operation defined only for matrices of the same dimension. The *p*-product is defined for matrices of possibly different dimensionality and extends the common notions of vector and matrix products. In general, we start with a heterogeneous algebra $\mathcal{A} = (\{\mathbb{E}, \mathbb{F}, \mathbb{G}\}, \{\bigcirc, \gamma\})$, where $\bigcirc$ is a binary operation $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$ and $(\mathbb{F}, \gamma)$ is a commutative semigroup. For each quadruple *l, m, n,* and *q* of positive integers and each fixed $p \in \mathbb{Z}^+$ that divides both *m* and *n,* we construct an induced heterogeneous matrix algebra $\mathcal{A}_p = \left(\left\{ M_{l\times m}(\mathbb{E}),\, M_{l\frac{n}{p}\times\frac{m}{p}q}(\mathbb{F}),\, M_{n\times q}(\mathbb{G})\right\}, \left\{ \textcircled{\gamma}_p,\, \gamma\right\}\right)$, where $\left(M_{l\frac{n}{p}\times\frac{m}{p}q}(\mathbb{F})\,, \gamma\right)$ is the induced matrix semigroup defined by Eq. 3.14.1 and

$$\textcircled{\gamma}_p : M_{l\times m}(\mathbb{E}) \times M_{n\times q}(\mathbb{G}) \to M_{l\frac{n}{p}\times\frac{m}{p}q}(\mathbb{F}) \tag{3.14.8}$$

is a binary operation induced by $\bigcirc$ and γ. The operation $\textcircled{\gamma}_p$ is called the *general matrix product of order p* or simply the *p-product* and is defined as follows: Suppose $A = (a_{s,t}) \in M_{l\times m}(\mathbb{E})$ and $B = (b_{s,t}) \in M_{n\times q}(\mathbb{G})$. Re-index *A* and *B* using the rule

$$\begin{aligned} A &= \left(a_{s,(i,k)}\right)_{l\times m} \;\Leftrightarrow\; r_p(i,k) = t, \text{ and} \\ B &= \left(b_{(k,j),t}\right)_{n\times q} \;\Leftrightarrow\; c_p(k,j) = s\,. \end{aligned}$$

We now define the matrix

$$C = A \textcircled{\gamma}_p B \in M_{l\frac{n}{p} \times \frac{m}{p} q}(\mathbb{F}) \tag{3.14.9}$$

by

$$c_{(s,j)(i,t)} = \mathop{\mathbf{\Gamma}}_{k=1}^{p} \left(a_{s,(i,k)} \bigcirc b_{(k,j),t}\right) = \left(a_{s,(i,1)} \bigcirc b_{(1,j),t}\right)\gamma \ldots \gamma\left(a_{s,(i,p)} \bigcirc b_{(p,j),t}\right), \tag{3.14.10}$$

where $c_{(s,j)(i,t)}$ is the entry in the (*s*,*j*)-row and (*i*,*t*)-column of C. Here we use the lexicographical order $(s,j) < (s',j') \Leftrightarrow s < s'$ or if $s = s'$, $j < j'$. Thus, the matrix C has the following form:

$$\begin{pmatrix}
c_{(1,1)(1,1)} & \cdots & c_{(1,1)(1,q)} & c_{(1,1)(2,1)} & \cdots & c_{(1,1)(2,q)} & \cdots & c_{(1,1)(i,t)} & \cdots & c_{(1,1)(m/p,q)} \\
c_{(1,2)(1,1)} & \cdots & c_{(1,2)(1,q)} & c_{(1,2)(2,1)} & \cdots & c_{(1,2)(2,q)} & \cdots & c_{(1,2)(i,t)} & \cdots & c_{(1,2)(m/p,q)} \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\
c_{(1,n/p)(1,1)} & \cdots & c_{(1,n/p)(1,q)} & c_{(1,n/p)(2,1)} & \cdots & c_{(1,n/p)(2,q)} & \cdots & c_{(1,n/p)(i,t)} & \cdots & c_{(1,n(p))(m/p,q)} \\
c_{(2,1)(1,1)} & \cdots & c_{(2,1)(1,q)} & c_{(2,1)(2,1)} & \cdots & c_{(2,1)(2,q)} & \cdots & c_{(2,1)(i,t)} & \cdots & c_{(2,1)(m/p,q)} \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\
c_{(2,n/p)(1,1)} & \cdots & c_{(2,n/p)(1,q)} & c_{(2,n/p)(2,1)} & \cdots & c_{(2,n/p)(2,q)} & \cdots & c_{(2,n/p)(i,t)} & \cdots & c_{(2,n/p)(m/p,q)} \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\
c_{(s,j)(1,1)} & \cdots & c_{(s,j)(1,q)} & c_{(s,j)(2,1)} & \cdots & c_{(s,j)(2,q)} & \cdots & \underline{c_{(s,j)(i,t)}} & \cdots & c_{(s,j)(m/p,q)} \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\
c_{(l,1)(1,1)} & \cdots & c_{(l,1)(1,q)} & c_{(l,1)(2,1)} & \cdots & c_{(l,1)(2,q)} & \cdots & c_{(l,1)(i,t)} & \cdots & c_{(l,1)(m/p,q)} \\
\vdots & & \vdots & \vdots & & \vdots & & \vdots & & \vdots \\
c_{(l,n/p)(1,1)} & \cdots & c_{(l,n/p)(1,q)} & c_{(l,n/p)(2,1)} & \cdots & c_{(l,n/p)(2,q)} & \cdots & c_{(l,n/p)(i,t)} & \cdots & c_{(l,n/p)(m/p,q)}
\end{pmatrix}$$

The entry $c_{(s,j)(i,t)}$ in the (*s*,*j*)-row and (*i*,*t*)-column is underlined for emphasis.

3.14.3 Example: Suppose $\mathbb{E} = \mathbb{F} = \mathbb{G} = \mathbb{R}$ and γ denotes addition (+), while $\bigcirc$ denotes multiplication in the above description. If $l = 2,\ m = 6,\ n = 4,\ \text{and}\ q = 3$, then for $p = 2$, one obtains $m/p = 3,\ n/p = 2\ \text{and}\ 1 \le k \le 2$. Now let

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \end{pmatrix} \in M_{l\times m}(\mathbb{R}) = M_{2\times 6}(\mathbb{R})$$

and

$$B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{pmatrix} \in M_{n\times q}(\mathbb{R}) = M_{4\times 3}(\mathbb{R}).$$

Then the (2,1)-row and (2,3)-column element $c_{(2,1)(2,3)}$ of the matrix

$$C = A \bigoplus_2 B \in M_{ln/p\times(m/p)q}(\mathbb{R}) = M_{4\times 9}(\mathbb{R})$$

is given by

$$\begin{aligned} c_{(2,1)(2,3)} &= \sum_{k=1}^{2} a_{2,(2,k)} \cdot b_{(k,1),3} \\ &= a_{2,(2,1)} \cdot b_{(1,1),3} + a_{2,(2,2)} \cdot b_{(2,1),3} \\ &= a_{23} \cdot b_{13} + a_{24} \cdot b_{33}. \end{aligned}$$

Thus, in order to compute $c_{(2,1)(2,3)}$, the two underlined elements of A are combined with the two underlined elements of B as illustrated:

$$
\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & \underline{a_{23}} & \underline{a_{24}} & a_{25} & a_{26} \end{pmatrix} \oplus_2 \begin{pmatrix} b_{11} & b_{12} & \underline{b_{13}} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & \underline{b_{33}} \\ b_{41} & b_{42} & b_{43} \end{pmatrix}
$$

$$
= \begin{pmatrix} a_{1,(1,1)} & a_{1,(1,2)} & a_{1,(2,1)} & a_{1,(2,2)} & a_{1,(3,1)} & a_{1,(3,2)} \\ a_{2,(1,1)} & a_{2,(1,2)} & \underline{a_{2,(2,1)}} & \underline{a_{2,(2,2)}} & a_{2,(3,1)} & a_{2,(3,2)} \end{pmatrix} \oplus_2 \begin{pmatrix} b_{(1,1),1} & b_{(1,1),2} & \underline{b_{(1,1),3}} \\ b_{(1,2),1} & b_{(1,2),2} & b_{(1,2),3} \\ b_{(2,1),1} & b_{(2,1),2} & \underline{b_{(2,1),3}} \\ b_{(2,2),1} & b_{(2,2),2} & b_{(2,2),3} \end{pmatrix}
$$

$$
= \begin{pmatrix} c_{(1,1)(1,1)} & c_{(1,1)(1,2)} & \cdots & c_{(1,1)(2,3)} & \cdots & c_{(1,1)(3,3)} \\ c_{(1,2)(1,1)} & c_{(1,2)(1,2)} & \cdots & c_{(1,2)(2,3)} & \cdots & c_{(1,2)(3,3)} \\ c_{(2,1)(1,1)} & c_{(2,1)(1,2)} & \cdots & \underline{c_{(2,1)(2,3)}} & \cdots & c_{(2,1)(3,3)} \\ c_{(2,2)(1,1)} & c_{(2,2)(1,2)} & \cdots & c_{(2,2)(2,3)} & \cdots & c_{(2,2)(3,3)} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{16} & \cdots & c_{19} \\ c_{21} & c_{22} & \cdots & c_{26} & \cdots & c_{29} \\ c_{31} & c_{32} & \cdots & \underline{c_{36}} & \cdots & c_{39} \\ c_{41} & c_{42} & \cdots & c_{46} & \cdots & c_{49} \end{pmatrix}.
$$

The notation $\textcircled{$\gamma$}_p$ is complete in that it involves all the building blocks of the *p*-product; the subscript *p* provides us with the common divisor of *m* and *n*, $\bigcirc$ reminds us of the binary operation $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$, and the operation γ inscribed by the circle defines the global reduce operation Γ of Eq. 3.14.10 of elements of $\mathbb{F}$. We make the convention of using a circle, $\bigcirc$, if the binary operation $\mathbb{E} \times \mathbb{G} \to \mathbb{F}$ is viewed as a multiplicative operation and a square, $\square$, if it is viewed as an additive operation. Thus if $\mathbb{E} = \mathbb{F} = \mathbb{G} = \mathbb{R}$ and γ denotes addition (+), while $\bigcirc$ denotes multiplication, then Eqs. 3.14.9 and 3.14.10 have form

$$
C = A \oplus_p B \in M_{l\frac{n}{p} \times \frac{m}{p} q}(\mathbb{R}) \tag{3.14.11}
$$

and

$$
c_{(s,j)(i,t)} = \sum_{k=1}^{p} \left(a_{s,r_p(i,k)} \cdot b_{c_p(k,j),t} \right) = \left(a_{s,r_p(i,1)} \cdot b_{c_p(1,j),t} \right) + \ldots + \left(a_{s,r_p(i,p)} \cdot b_{c_p(p,j),t} \right), \tag{3.14.12}
$$

respectively. On the other hand, if γ denotes the maximum, $\vee$, of two numbers and $\bigcirc$ denotes addition, then Eqs. 3.14.9 and 3.14.10 have form

$$
C = A \boxed{\vee}_p B \in M_{l\frac{n}{p} \times \frac{m}{p} q}(\mathbb{R}) \tag{3.14.13}
$$

and

$$
c_{(s,j)(i,t)} = \bigvee_{k=1}^{p} \left(a_{s,(i,k)} + b_{(k,j),t} \right) = \left(a_{s,(i,1)} + b_{(1,j),t} \right) \vee \ldots \vee \left(a_{s,(i,p)} + b_{(p,j),t} \right), \tag{3.14.14}
$$

respectively.

As mentioned earlier, the *p*-product includes the commonly used matrix and vector products. These products are obtained when substituting certain specific values for *p*. We conclude this section by considering these specific cases.

3.14.4 The case $p = 1$.

If $p = 1$, then $m/p = m$, $n/p = n$ and $k = 1$. Therefore, the entry in the (*s*,*j*)-row and (*i*,*t*)-column of the matrix $C = A \circledgamma_1 B \in M_{ln \times mq}(\mathbb{F})$ is given by $c_{(s,j)(i,t)} = a_{s,(i,1)} \bigcirc b_{(1,j),t}$. Since $r_1(i,1) = i$ and $c_1(1,j) = j$, $c_{(s,j)(i,t)}$ is of form $c_{(s,j)(i,t)} = a_{si} \bigcirc b_{jt}$. Hence,

$$A \circledgamma_1 B = \begin{pmatrix} a_{11} \bigcirc B & a_{12} \bigcirc B & \cdots & a_{1m} \bigcirc B \\ \vdots & \vdots & & \vdots \\ a_{l1} B & a_{l2} \bigcirc B & \cdots & a_{lm} \bigcirc B \end{pmatrix},$$

where

$$a_{si} \bigcirc B = \begin{pmatrix} a_{si} \bigcirc b_{11} & \cdots & a_{si} \bigcirc b_{1q} \\ \vdots & & \vdots \\ a_{si} \bigcirc b_{n1} & \cdots & a_{si} \bigcirc b_{nq} \end{pmatrix}.$$

This is the *Kronecker* or *tensor* product for heterogeneous algebras. For this reason, whenever γ and $\bigcirc$ are understood, we let $\circledgamma_1$ be denoted by $\otimes$.

If both $p = 1$ and $m = 1$, then A is a column vector of length l and

$$A \otimes B = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_l \end{pmatrix} \circledgamma_1 B = \begin{pmatrix} a_1 \bigcirc B \\ a_2 \bigcirc B \\ \vdots \\ a_l \bigcirc B \end{pmatrix} \in M_{ln \times q}(\mathbb{F}),$$

which corresponds to the tensor product of a column vector and a matrix.

If instead of m, $n = 1$, then we obtain the heterogeneous version of the tensor product of a row vector and a matrix:

$$(a_1, a_2, \cdots, a_m) \otimes B = (a_1 \bigcirc B, a_2 \bigcirc B, \cdots, a_m \bigcirc B) \in M_{n \times mq}(\mathbb{F}).$$

If both $l = 1$ and $q = 1$, then we obtain the tensor product of a row vector and a column vector:

$$\begin{aligned} A \otimes B &= (a_1 \bigcirc B, a_2 \bigcirc B, \cdots, a_m \bigcirc B). \\ &= \begin{pmatrix} a_1 b_1 & a_2 b_1 & \cdots & a_m b_1 \\ a_1 b_2 & a_2 b_2 & \cdots & a_m b_2 \\ \vdots & \vdots & & \vdots \\ a_1 b_n & a_2 b_n & \cdots & a_m b_n \end{pmatrix} \in M_{n \times m}(\mathbb{F}). \end{aligned}$$

Observe that if $m = n$, then this general tensor product corresponds to the *outer product* of two row vectors.

If $m = 1$ and $n = 1$, then $A \in M_{l \times 1}(\mathbb{E})$, $B \in M_{1 \times q}(\mathbb{G})$ and

$$\begin{aligned} A \otimes B &= \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_l \end{pmatrix} \circledgamma_1 B = \begin{pmatrix} a_1 \bigcirc B \\ a_2 \bigcirc B \\ \vdots \\ a_l \bigcirc B \end{pmatrix} \\ &= \begin{pmatrix} a_1 b_1 & a_1 b_2 & \cdots & a_1 b_q \\ a_2 b_1 & a_2 b_2 & \cdots & a_2 b_q \\ \vdots & \vdots & & \vdots \\ a_l b_1 & a_l b_2 & \cdots & a_l b_q \end{pmatrix} \in M_{l \times q}(\mathbb{F}). \end{aligned}$$

It follows from the last two cases that if $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$ is commutative, then $(B' \otimes A')' = A \otimes B$, where $A' \in M_{1\times l}(\mathbb{E})$ and $B' \in M_{q\times 1}(\mathbb{G})$ are the transpose of $A \in M_{l\times 1}(\mathbb{E})$ and $B \in M_{1\times q}(\mathbb{G})$, respectively.

If $l = 1 = n$ and $m = q$, then $r_1(i,k) = r_1(i,1) = i$, $c_1(k,j) = c_1(1,1) = 1$, and $c_{(s,j)(i,t)} = c_{(1,1)(i,t)} = c_{1,(i,t)} = c_{(i,t)}$, where $c_{(i,t)} = a_i \bigcirc b_t$. Thus,

$$\begin{aligned} A \otimes B = C &= \left(c_{(1,1)},\, c_{(1,2)},\, \cdots,\, c_{(1,m)},\, c_{(2,1)},\, \cdots, c_{(2,m)},\, \cdots,\, c_{(m,m)}\right) \\ &= (c_1,\, c_2,\, \cdots,\, c_m,\, c_{m+1},\, \cdots,\, c_{2m},\, \cdots,\, c_{m^2}) \end{aligned}$$

and $\otimes : \mathbb{E}^m \times \mathbb{G}^m \to \mathbb{F}^{m^2}$.

3.14.5 The case $p = m = n$.

In this case $m/p = n/p = 1$. Since $1 \le i \le m/p = 1$ and $1 \le j \le n/p = 1$, we have that $1 \le r_p(i,k) = k = c_p(k,j) \le m$. Therefore, $\textcircled{\gamma}_m : M_{l\times m}(\mathbb{E}) \times M_{m\times q}(\mathbb{G}) \to M_{l\times q}(\mathbb{F})$, where $C = A \textcircled{\gamma}_m B$ is defined by

$$c_{st} = \mathop{\Gamma}_{k=1}^{m} a_{sk} \bigcirc b_{kt}\,. \tag{3.14.15}$$

Here we set $c_{(s,j)(i,t)} = c_{st}$ since $(s,j) = (s,1) = s$ and $(i,t) = (1,t) = t$. We also define $\textcircled{\gamma} = \textcircled{\gamma}_p$ whenever $m = n = p$.

Replacing Γ with Σ, we see that 3.14.15 has the form of the usual matrix product. For this reason we refer to $\textcircled{\gamma}$ as the *generalized matrix product*.

If $l = 1$, then $\textcircled{\gamma} : \mathbb{E}^m \times M_{m\times q}(\mathbb{G}) \to \mathbb{F}^q$ and $C = A \textcircled{\gamma} B$ is defined by

$$c_t = \mathop{\Gamma}_{k=1}^{m} a_k \bigcirc b_{kt} \tag{3.14.16}$$

since in this case $c_{(s,j)(i,t)} = c_{(1,1)(1,t)} = c_t$ and $1 \le r_p(i,k) = r_p(1,k) = k = c_p(k,1) = c_p(k,j) \le m$.

An analogous case occurs when $q = 1$. In this case $\textcircled{\gamma} : M_{l\times m}(\mathbb{E}) \times (\mathbb{G}^m)' \to \left(\mathbb{F}^l\right)'$ and $C = A \textcircled{\gamma} B$ is defined by

$$c_s = \mathop{\Gamma}_{k=1}^{m} a_{sk} \bigcirc b_k\,. \tag{3.14.17}$$

Equations 3.14.15, 3.14.16, and 3.14.17 play a central role in image algebra and correspond to the general template product, the general backward transform (or backward image-template product), and the general forward transform (or forward image-template product), respectively.

Another important product occurs whenever $l = q = 1$. In this case $\textcircled{\gamma} : \mathbb{E}^m \times \mathbb{G}^m \to \mathbb{F}$. Thus, $C = A \textcircled{\gamma} B$ is a *scalar* value in $\mathbb{F}$. Since $c_{(s,j)(i,t)} = c_{(1,1)(1,1)} = c_{11}$, we set $c = c_{(s,j)(i,t)}$ and note that

$$c = \mathop{\Gamma}_{k=1}^{m} a_k \bigcirc b_k\,. \tag{3.14.18}$$

Again, replacing Γ with Σ, we see that 3.14.18 has the form of the usual *inner* or *dot product* of two vectors. For this reason we call $A \textcircled{\gamma} B$ the *generalized dot product* and set $A \textcircled{\gamma} B = A \bullet B$.

Bibliography

[1] R.C. Backhouse and B. Carré. Regular algebra applied to path-finding problems. *J. Inst. Math. Appl.*, 15:161–186, 1975.

[2] C. Benzaken. Structures algébra des cheminements. In G. Biorci, editor, *Network and Switching Theory*, pages 40–57. Academic Press, 1968.

[3] G. Birkhoff. *Lattice Theory*. American Mathematical Society, Providence, RI, 1984.

[4] G. Birkhoff and J. Lipson. Heterogeneous algebras. *Journal of Combinatorial Theory*, 8:115–133, 1970.

[5] R. Blahut. *Fast Algorithms for Digital Signal Processing*. Addison-Wesley, Reading, MA, 1985.

[6] B. Carré. An algebra for network routing problems. *J. Inst. Math. Appl.*, 7:273–294, 1971.

[7] R. Cuninghame-Green. Process synchronisation in steelworks—a problem of feasibility. In Banbury and Maitland, editors, *Procceedings of the 2nd International Conference on Operations Research*, pages 323–328. English University Press, 1960.

[8] R. Cuninghame-Green. Describing industrial processes with interference and approximating their steady-state behaviour. *Oper. Research Quart.*, 13:95–100, 1962.

[9] R. Cuninghame-Green. *Minimax Algebra: Lecture Notes in Economics and Mathematical Systems 166*. Springer-Verlag, New York, 1979.

[10] J.L. Davidson. *Lattice Structures in the Image Algebra and Applications to Image Processing*. PhD thesis, University of Florida, Gainesville, FL, 1989.

[11] R. Dedekind. *Gesammelte Mathematische Werke*. Math. Annalen, Braunschweig, 1927.

[12] P.D. Gader. *Image Algebra Techniques for Parallel Computation of Discrete Fourier Transforms and General Linear Transforms*. PhD thesis, University of Florida, Gainesville, FL, 1986.

[13] B. Giffler. Mathematical solution of production planning and scheduling problems. Tech. rep., IBM ASDD, 1960.

[14] P.J. Higgins. Algebras with a scheme of operators. *Mathematische Nachrichten*, 27:115–132, 1963.

[15] F. Klein. *Gesammelte mathematische Abhandlungen*, volume 1. Julius Springer, Goettingen-Berlin, 1921.

[16] S. Lang. *Algebra*. Addison-Wesley, Reading, MA, 1965.

[17] Jr M. Hall. *The Theory of Groups*. Macmillan Co., New York, 1966.

[18] V. Peteanu. An algebra of the optimal path in networks. *Mathematica*, 9:335–342, 1967.

[19] G.X. Ritter and J.L. Davidson. The image algebra and lattice theory. Technical Report TR 87–09, University of Florida CIS Department, 1987.

[20] W. Rudin. *Real and Complex Analysis*. McGraw-Hill, New York, NY, 1974.

[21] A. Shimbel. Structure in communication nets. In *Proceedings of the Symposium on Information Networks*, pages 119–203. Polytechnic Institute of Brooklyn, 1954.

[22] D. Wenzel. Adaption of group algebras to image processing. Master's thesis, University of Texas at San Antonio, Dept. of Computer Science, December 1988.

CHAPTER 4
IMAGE ALGEBRA

The goal of this chapter is to familiarize the reader with the basic concepts that define image algebra. Since the primary operands of image algebra are images and templates, we begin our discussion with their characterization.

4.1 Images and Templates

In the design and analysis of computer vision algorithms it is not only convenient, but also necessary to mathematically characterize the images to be manipulated.

In Section **2.18** we discussed the notion of a digital image. According to Definition **2.18.3**, digital images are quantized versions of continuous intensity functions over some spatial domain **X**. Specifically, digital images were defined as functions with domain a rectangular subset of $\mathbb{Z}^2$ and range some subset of $\mathbb{Z}_{2^k}$. Although most digital images assume this type of format, the definition does not cover various types of images manipulated by current digital and optical devices. For instance, digital computers have two major forms of numeric representations: integer and real. Integer numbers range from 0 to some maximum value $2^k - 1$. In a 32–bit computer, for example, the maximum positive integer is $2^{32} - 1$. If an integer arithmetic operation results in a fractional part, the remainder is simply truncated. Thus, for example, the ratio $\frac{16}{3}$ is represented as the integer 5 without a trailing decimal point. With real number computation, the fractional part of an operation is retained up to the numerical accuracy of the computer. The ratio of the real numbers $\frac{16.0}{3.0}$ is represented as $5.33 \cdots 33$. Most image processing techniques involve real or floating point arithmetic, thus necessitating real-valued image representation. Furthermore, various image transformations, such as edge detection transforms and the Fourier transform, introduce negative and complex values. Also, when modeling images and image transforms in the continuous domain, a continuum of values is usually required for the range of an image function. Thus, the set $\mathbb{Z}_{2^k}$ does not suffice to characterize image values and image processing transforms.

Three-dimensional images, i.e., images with domain in $\mathbb{Z}^3$ or $\mathbb{R}^3$, are often generated (computed) from multiple camera views, motion, millimeter wave radar (MMWR), X-ray tomography, or laser-range data. Therefore, the need for domains other than rectangular subsets of $\mathbb{Z}^2$ is also obvious. In order to provide a mathematically rigorous definition of an image that covers this plethora of different images, it becomes evident that an image must be defined in general terms, with minimum specification.

4.1.1 **Definition.** Let $\mathbb{F}$ be a homogeneous algebra and **X** a topological space. An $\mathbb{F}$*–valued image on* **X** is any element of $\mathbb{F}^{\mathbf{X}}$. Given an $\mathbb{F}$–valued image $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$, then $\mathbb{F}$ is called the *set of possible range values* of **a** and **X** the *spatial domain* of **a**.

It is often convenient to let the graph of an image $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ represent **a**. The graph of an image is also referred to as the *data structure* representation of the image. Given the data structure representation $\mathbf{a} = \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{x} \in \mathbf{X}\}$, then an element $(\mathbf{x}, \mathbf{a}(\mathbf{x}))$ of the data structure is called a *picture element* or *pixel*. The first coordinate **x** of a pixel is called the *pixel location* or *image point*, and the second coordinate **a**(**x**) is called the *pixel value* of **a** at location **x**.

Obviously, Definition **4.1.1** covers all mathematical images on topological spaces with range in an algebraic system. Requiring **X** to be a topological space provides us with the notion of nearness of pixels. Since **X** is not directly specified we may substitute any space required for the analysis of an image or imposed by a particular sensor and scene. For example, **X** could be a subset of $\mathbb{Z}^3$ or $\mathbb{R}^3$ with $\mathbf{x} \in \mathbf{X}$ of form $\mathbf{x} = (x, y, t)$, where the first coordinates (x, y) denote spatial location and *t* a time variable.

Similarly, replacing the unspecified value set $\mathbb{F}$ with $\mathbb{Z}_{2^k}$ or $\mathbb{F} = (\mathbb{Z}_{2^k}, \mathbb{Z}_{2^m}, \mathbb{Z}_{2^n})$ provides us with digital integer-valued and digital vector-valued images, respectively. An implication of these observations is that **4.1.1** also characterizes any type of discrete or continuous *physical* image. The common model of a physical image is in terms of a continuous energy function $E(\mathbf{x}, \lambda)$, where λ is a variable associated with energy at space/time location **x**. In the context of visual images λ refers to wavelength and **x** may be of form $\mathbf{x} = (x, y)$, $\mathbf{x} = (x, y, t)$, or $\mathbf{x} = (x, y, z, t)$, where (x, y) and (x, y, z) represent spatial coordinates and *t* the time variable. Physical imaging systems impose a restriction on the maximum brightness of an image and since light intensity is a real positive quantity, it is assumed that

$$0 \leq E(\mathbf{x}, \lambda) \leq B\,,$$

where *B* denotes the maximum brightness. In addition, the spatial domain is limited in extent by the imaging system and the system operates only over a finite time interval so that $|\mathbf{x}| < L$. Thus, *E* is a bounded function on $\mathbf{X} \times [0, \infty)$, where **X** is compact, $\mathbf{x} \in \mathbf{X}$, and $\lambda \in [0, \infty)$. In many imaging systems the image does not change with time so that **X** represents only the set of spatial domain variables.

The observed image field is modeled as a spectrally weighted interval

$$\mathbf{a}(\mathbf{x}) = \int_0^\infty E(\mathbf{x}, \lambda) s(\lambda) d\lambda\,,$$

where $s(\lambda)$ denotes the spectral response of the sensor used. In a multispectral imaging system the observed image field is given by

$$\mathbf{a}(\mathbf{x}) = (\mathbf{a}_1(\mathbf{x}), \ldots, \mathbf{a}_n(\mathbf{x}))\,,$$

where

$$\mathbf{a}_i(\mathbf{x}) = \int_0^\infty \mathrm{E}(\mathbf{x}, \lambda) s_i(\lambda) d\lambda\,,$$

and $s_i(\mathbf{x})$ denotes the response of the *i*th sensor. For example, in an arbitrary red-green-blue coordinate system,

$$\mathbf{a}(\mathbf{x}) = (\mathbf{r}(\mathbf{x}), \mathbf{g}(\mathbf{x}), \mathbf{b}(\mathbf{x}))\,,$$

where

$$\mathbf{r}(\mathbf{x}) = \int_0^\infty E(\mathbf{x}, \lambda) s_1(\lambda) d\lambda\,,$$

$$\mathbf{g}(\mathbf{x}) = \int_0^\infty E(\mathbf{x}, \lambda) s_2(\lambda) d\lambda\,,$$

$$\mathbf{b}(\mathbf{x}) = \int_0^\infty E(\mathbf{x}, \lambda) s_3(\lambda) d\lambda\,,$$

and $s_1(\lambda)$, $s_2(\lambda)$, and $s_3(\lambda)$ are spectral tristimulus values for the set of red, green, and blue primaries, respectively.

Although these examples are presented in the context of visual images, other multidimensional time varying signals have similar representations and can always be viewed as bounded functions from a compact set **X** to an appropriate value set $\mathbb{F}$. The sampled (quantized) versions of these functions remain, of course, bounded functions on compact sets as finite discrete sets are always compact. Therefore, both continuous and quantized image representations of physical images are included in Definition **4.1.1**.

Even though Definition **4.1.1** provides a general framework for specifying images with particular point and value sets, the definition does *not* provide for more insight into understanding image content; the functional form of the image **a** is almost never known. The inability to relate changes in **x** to those in **a**(**x**) presents one of the greatest difficulties in image understanding. In addition, **4.1.1** provides only for a deterministic image representation; it defines a mathematical image function with point properties. It is often convenient to regard an image as a sample of a stochastic process. The image function $\mathbf{a}(\mathbf{x}) = \mathbf{a}(x, y, z, t)$ (or $\mathbf{a}(\mathbf{x}) = \mathbf{a}(x, y, t)$) is assumed to be a member of a continuous four-dimensional stochastic process with space variable (x, y, z) and time variable *t*. A discrete (quantized) image array can be completely characterized statistically by its joint probability density. The extension of image algebra to include stochastic models is given in Section **?**.

A more abstract class of images is provided by templates. Templates are images whose *values* are images. In terms of image processing applications, templates reign supreme; template operations play a dominant role in most algorithms and provide for brevity of code. They are the essence of image algebra. The notion of a template, as used in image algebra, unifies and generalizes the usual concepts of templates, masks, windows, and neighborhood functions into one general mathematical entity. In addition, templates generalize the notion of structuring elements as used in mathematical morphology [5, 4].

4.1.2 **Definition.** A *template* is an image whose pixel values are images (functions). In particular, an $\mathbb{F}$–*valued template from* $\mathbf{Y}$ *to* $\mathbf{X}$ is a function $\mathbf{t} : \mathbf{Y} \to \mathbb{F}^{\mathbf{X}}$. Thus, $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ and **t** is an $\mathbb{F}^{\mathbf{X}}$–valued image on **Y**.

For notational convenience we define $\mathbf{t_y} \equiv \mathbf{t}(\mathbf{y})\ \forall \mathbf{y} \in \mathbf{Y}$. The image $\mathbf{t_y}$ has representation

$$\mathbf{t_y} = \{(\mathbf{x}, \mathbf{t_y}(\mathbf{x})) : \mathbf{x} \in \mathbf{X}\}.$$

The pixel values $\mathbf{t_y}(\mathbf{x})$ of this image are called the *weights* of the template at point **y**.

From a set theoretic point of view the set $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ of all $\mathbb{F}$–valued templates from $\mathbf{Y}$ to $\mathbf{X}$ and the set $\mathbb{F}^{\mathbf{X}\times\mathbf{Y}}$ are equivalent. Defining for each $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ a corresponding function $\hat{\mathbf{t}} \in \mathbb{F}^{\mathbf{X}\times\mathbf{Y}}$ by $\hat{\mathbf{t}}(\mathbf{x}, \mathbf{y}) = \mathbf{t_y}(\mathbf{x})$, and vice versa, provides for the necessary one-to-one correspondence. Conceptually, however, there is a great distinction between the elements of $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ and those of $\mathbb{F}^{\mathbf{X}\times\mathbf{Y}}$. For the former we obtain for each point $\mathbf{y} \in \mathbf{Y}$ an $\mathbb{F}$–valued image $\mathbf{t_y}$ on $\mathbf{X}$ while for the latter we obtain for each point $(\mathbf{x}, \mathbf{y}) \in \mathbf{X} \times \mathbf{Y}$ a value $\mathbf{t}(\mathbf{x}, \mathbf{y}) \in \mathbb{F}$. It is this concept of associating with each point $\mathbf{y} \in \mathbf{Y}$ an image $\mathbf{t_y}$ that provides templates with their great utility.

If **t** is a real or complex-valued template from **Y** to **X**, then the *support* of $\mathbf{t_y}$ is denoted by $S(\mathbf{t_y})$ and is defined as

$$S(\mathbf{t_y}) = \{\mathbf{x} \in \mathbf{X} : \mathbf{t_y}(\mathbf{x}) \neq 0\}.$$

For extended real-valued templates we also define the following supports *at infinity*:

$$S_{\infty}(\mathbf{t_y}) = \{\mathbf{x} \in \mathbf{X} : \mathbf{t_y}(\mathbf{x}) \neq \infty\},$$
$$S_{-\infty}(\mathbf{t_y}) = \{\mathbf{x} \in \mathbf{X} : \mathbf{t_y}(\mathbf{x}) \neq -\infty\},$$

and

$$S_{\pm\infty}(\mathbf{t_y}) = \{\mathbf{x} \in \mathbf{X} : \mathbf{t_y}(\mathbf{x}) \neq \pm\infty\}.$$

If **X** is a space with an operation + such that $(\mathbf{X}, +)$ is a group, then a template $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{X}}$ is said to be *translation invariant* (with respect to the operation +) if and only if for each triple $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbf{X}$ we have that $\mathbf{t_y}(\mathbf{x}) = \mathbf{t_{y+z}}(\mathbf{x} + \mathbf{z})$. Templates that are not translation invariant are called *translation variant* or, simply, *variant* templates. A large class of translation invariant templates with finite support have the nice property that they can be defined pictorially. For example, let $\mathbf{X} = \mathbb{Z}^2$ and **y** = (*x*,*y*) be an arbitrary point of **X**. Set $\mathbf{x}_1 = (x, y-1)$, $\mathbf{x}_2 = (x+1, y)$, and $\mathbf{x}_3 = (x+1, y-1)$. Define $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{X}}$ by defining the weights $\mathbf{t_y}(\mathbf{y}) = 1$, $\mathbf{t_y}(\mathbf{x_1}) = 3$, $\mathbf{t_y}(\mathbf{x}_2) = 2$, $\mathbf{t_y}(\mathbf{x}_3) = 4$, and $\mathbf{t_y}(\mathbf{x}) = 0$ whenever **x** is not an element of $\{\mathbf{y}, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$. Note that it follows from the definition of **t** that $S(\mathbf{t_y}) = \{\mathbf{y}, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$. Thus, at any arbitrary point **y**, the configuration of the support and weights of $\mathbf{t_y}$ is as shown in Figure **4.1.1**. The shaded cell in the pictorial representation of $\mathbf{t_y}$ indicates the location of the point **y**.

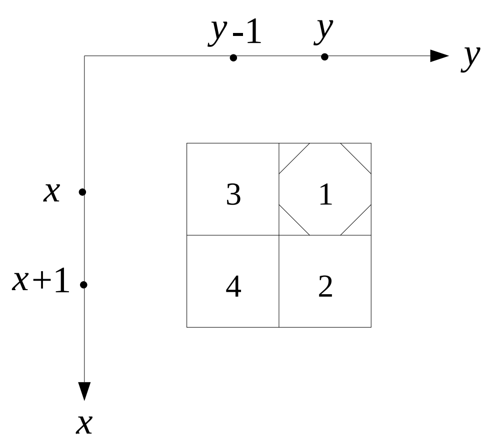

Figure 4.1.1 Pictorial representation of a translation invariant template

There are certain collections of templates that can be defined explicitly in terms of parameters. These parameterized templates are of great practical importance.

4.1.3 **Definition.** A *parameterized* $\mathbb{F}$*–valued template from* **Y** *to* **X** *with parameters in P* is a function of form $\mathbf{t} : P \to \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$. The set *P* is called the *set of parameters* and each $p \in P$ is called a *parameter* of **t**.

Thus, a parameterized $\mathbb{F}$–valued template from **Y** to **X** gives rise to a family of regular $\mathbb{F}$–valued templates from **Y** to **X**, namely $\left\{\mathbf{t}(p) \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}} : p \in P\right\}$.

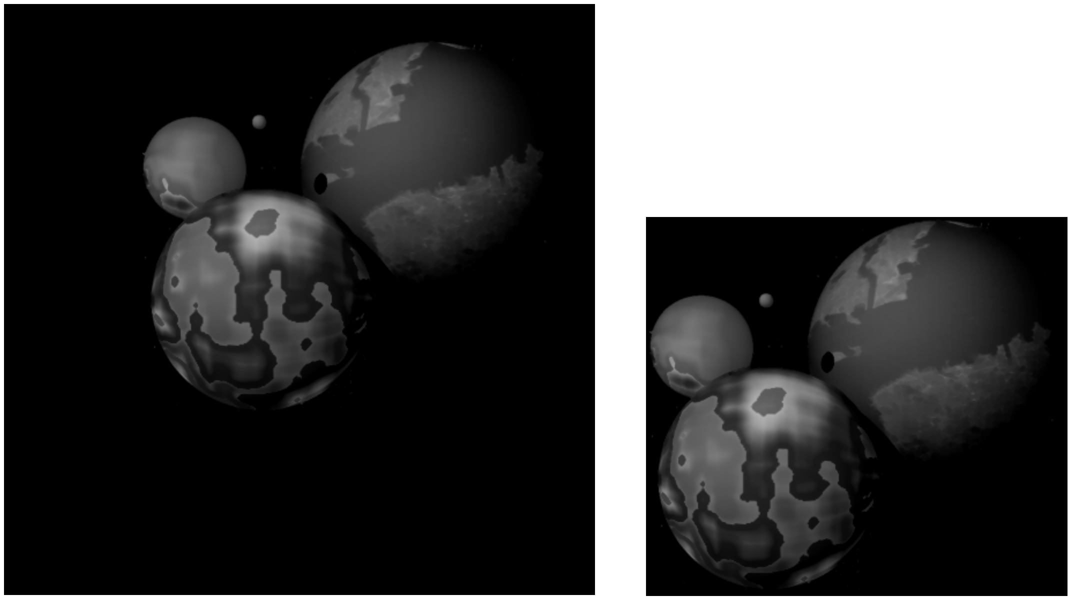

Figure 4.2.1 An example of an explicitly specified coordinate restriction.

4.2 Functional Specification of Image Processing Techniques

The basic concepts of elementary function theory discussed in Sections **2.5** and **2.6** provide the underlying foundation of a functional specification of image processing techniques. This is a direct consequence of viewing images as functions; since images are functions, the notation and concepts discussed in **2.5**, **2.6**, and subsequent sections can be used to furnish the functional notation for specifying image processing algorithms. In order to emphasize the fact that even the most elementary concepts of function theory have important applications in image processing, we begin the development of a functional notation by focusing our attention on the notions of domain, range, restriction, and extension of a function.

Image restrictions and extensions are used to restrict images to regions of particular interest and to embed images into larger images, respectively. Following the notation used in Section **2.5**, the *restriction* of $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ to a subset $\mathbf{Z}$ of $\mathbf{X}$ is denoted by $\mathbf{a}|_{\mathbf{Z}}$, and defined by $\mathbf{a}|_{\mathbf{Z}} = \mathbf{a} \cap (\mathbf{Z} \times \mathbb{F})$. Thus, $\mathbf{a}|_{\mathbf{Z}} \in \mathbb{F}^{\mathbf{Z}}$. In practice, the user may specify $\mathbf{Z}$ explicitly by providing bounds for the coordinates of the points of $\mathbf{Z}$. For example, the image $\mathbf{a}$ on the left of Figure **4.2.1** represents a digital image on the set $\mathbf{X} = \big\{(i,j) \in \mathbb{Z}^2 \,:\, 1 \le i \le 512,\ 1 \le j \le 512\big\}$, while the image on the right represents the restriction $\mathbf{a}|_{\{(i,j)\in\mathbf{X}\,:\,85\le i\le 485,\ 15\le j\le 380\}}$.

There is nothing magical about restricting $\mathbf{a}$ to a subset $\mathbf{Z}$ of its domain $\mathbf{X}$. We can just as well define restrictions of images to subsets of the range values. Specifically, if $S \subset \mathbb{F}$ and $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$, then the *restriction* of $\mathbf{a}$ to S is denoted by $\mathbf{a}||_S$ and defined as

$$\mathbf{a}||_S = \mathbf{a} \cap (\mathbf{X} \times S)\,.$$

In terms of the pixel representation of $\mathbf{a}||_S$ we have $\mathbf{a}||_S = \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) \,:\, \mathbf{a}(\mathbf{x}) \in S\}$. The double-bar notation is used to focus attention on the fact that the restriction is applied to the second coordinate of $\mathbf{a} \subset \mathbf{X} \times \mathbb{F}$.

Image restrictions in terms of subsets of the value set $\mathbb{F}$ is an extremely useful concept in computer vision as many image processing tasks are restricted to image domains over which the image values

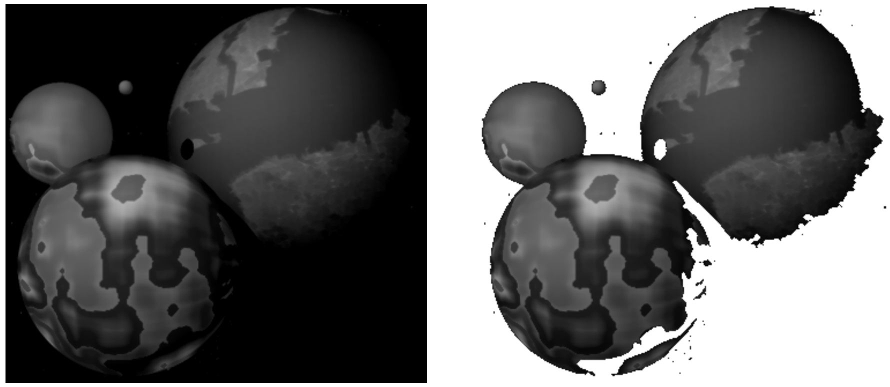

Figure 4.2.2 Example of a range restriction.

satisfy certain properties. Of course, one can always write this type of restriction in terms of a first coordinate restriction by setting $\mathbf{Z} = \{\mathbf{x} \in \mathbf{X} : \mathbf{a}(\mathbf{x}) \in S\}$ so that $\mathbf{a}||_S = \mathbf{a}|_{\mathbf{Z}}$. However, writing a program statement such as $\mathbf{b} := \mathbf{a}|_{\mathbf{Z}}$ is of little value since **Z** is implicitly specified in terms of *S*; i.e., **Z** must be determined in terms of the property "$\mathbf{a}(\mathbf{x}) \in S$." Thus, **Z** would have to be precomputed, adding to the computational overhead as well as increased code. In contrast, direct restriction of the second coordinate values to an explicitly specified set *S* avoids these problems and provides for easier implementation.

As mentioned, restrictions to the range set provide a useful tool for expressing various algorithmic procedures. For instance, if $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$ and *S* is the interval $(k, \infty) \subset \mathbb{R}$, where *k* denotes some given threshold value, then $\mathbf{a}||_{(k,\infty)}$ denotes the image **a** restricted to all those points of **X** where **a**(**x**) exceeds the value *k*. In order to reduce notation, we define $\mathbf{a}||_{>k} \equiv \mathbf{a}||_{(k,\infty)}$. Similarly,

$$\mathbf{a}||_{\geq k} \equiv \mathbf{a}||_{[k,\infty)}, \;\; \mathbf{a}||_{<k} \equiv \mathbf{a}||_{(-\infty,k)}, \;\; \mathbf{a}||_{k} \equiv \mathbf{a}||_{\{k\}}, \;\; \text{and} \;\; \mathbf{a}||_{\leq k} \equiv \mathbf{a}||_{(-\infty,k]}\,.$$

Figure **4.2.2** provides an example of such a restriction. In this example, the input **a** is shown on the left (it is identical to the explicitly restricted image shown on the right of Figure **4.2.1**), while the implicitly specified image $\mathbf{a}||_{>5}$ is shown on the right.

A more general form of range restriction is given when *S* corresponds to a set-valued image $S \in \left(2^{\mathbb{F}}\right)^{\mathbf{X}}$; i.e., $S(\mathbf{x}) \subset \mathbb{F} \;\; \forall \mathbf{x} \in \mathbf{X}$. In this case we define

$$\mathbf{a}||_S = \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) \in S(\mathbf{x})\}\,.$$

For example, given an image $\mathbf{b} \in \mathbb{R}^{\mathbf{X}}$, we may define a function $S_{\leq \mathbf{b}} : \mathbf{X} \to 2^{\mathbb{R}}$ by $S_{\leq \mathbf{b}}(\mathbf{x}) = \{r \in \mathbb{R} : r \leq \mathbf{b}(\mathbf{x})\}$. Then $\mathbf{a}||_{S_{\leq \mathbf{b}}} = \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) \leq \mathbf{b}(\mathbf{x})\}$. Again, in order to reduce notational baggage, we define $\mathbf{a}||_{\leq \mathbf{b}} \equiv \mathbf{a}||_{S_{\leq \mathbf{b}}}$. Similarly, we define

$$\begin{gathered}\mathbf{a}||_{<\mathbf{b}} \equiv \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) < \mathbf{b}(\mathbf{x})\}, \;\; \mathbf{a}||_{\mathbf{b}} \equiv \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) = \mathbf{b}(\mathbf{x})\}, \\ \mathbf{a}||_{\geq \mathbf{b}} \equiv \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) \geq \mathbf{b}(\mathbf{x})\}, \;\; \mathbf{a}||_{>\mathbf{b}} \equiv \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) > \mathbf{b}(\mathbf{x})\}, \\ \text{and} \;\; \mathbf{a}||_{\neq \mathbf{b}} \equiv \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{a}(\mathbf{x}) \neq \mathbf{b}(\mathbf{x})\}\,.\end{gathered}$$

Combining the concepts of first and second coordinate (domain and range) restrictions provides the general definition of an image restriction.

4.2.1 **Definition.** If $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$, $\mathbf{Z} \subset \mathbf{X}$, and $S \subset \mathbb{F}$, then the *restriction of* **a** *to* **Z** *and S* is defined as

$$\mathbf{a}|_{(\mathbf{Z},S)} = \mathbf{a} \cap (\mathbf{Z} \times S)\,.$$

It follows that $\mathbf{a}|_{(\mathbf{Z},S)} = \{(\mathbf{x}, \mathbf{a}(\mathbf{x})) : \mathbf{x} \in \mathbf{Z} \text{ and } \mathbf{a}(\mathbf{x}) \in S\}$, $\mathbf{a}|_{(\mathbf{X},S)} = \mathbf{a}||_S$, and $\mathbf{a}|_{(\mathbf{Z},\mathbb{F})} = \mathbf{a}|_{\mathbf{Z}}$.

The *extension* of $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ to $\mathbf{b} \in \mathbb{F}^{\mathbf{Y}}$ on **Y**, where **X** and **Y** are subsets of the same topological space, is denoted by $\mathbf{a}|^{\mathbf{b}}$ and defined by

$$\mathbf{a}|^{\mathbf{b}}(\mathbf{x}) = \begin{cases} \mathbf{a}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbf{X} \\ \mathbf{b}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbf{Y} \backslash \mathbf{X}\,. \end{cases}$$

In actual practice, the user will have to specify the function **b**. Figure **4.2.3** provides an example of such an extension. Here we extended the image $\mathbf{c} = \mathbf{a}||_{>5}$ (displayed on the right of Fig. **4.2.2**) with domain $\mathbf{Z} = \{\mathbf{x} \in \mathbf{X} : \mathbf{a}(\mathbf{x}) > 5\}$ to the image $\mathbf{d} = \mathbf{c}|^{\mathbf{b}}$ shown on the right. In this example, $\mathbf{Y} = \{(i,j) \in \mathbb{Z}^2 : 1 \le i \le 512,\ 1 \le j \le 512\}$, and **b** is the image displayed on the left of Fig. **4.2.3**.

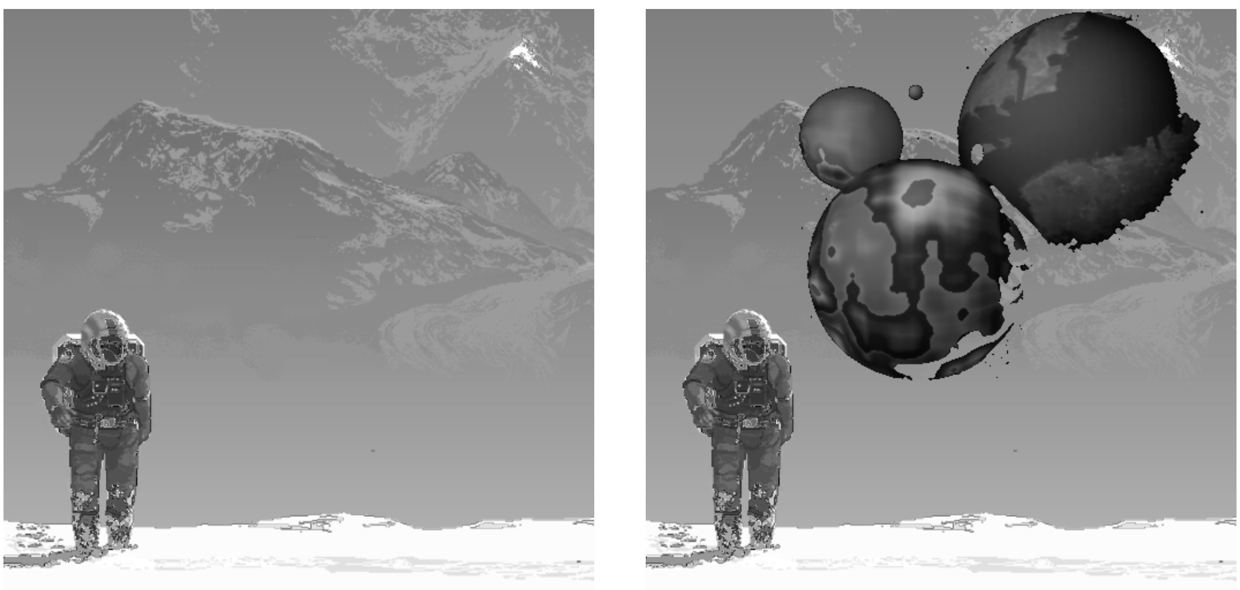

Figure 4.2.3 An example of an image extension.

Two of the most important concepts associated with a function are its domain and range (Section **2.6**). In the field of image understanding, it is convenient to view these concepts as functions that map images to sets associated with certain image properties. Specifically, we view the concept of range as a function

$$range : \mathbb{F}^{\mathbf{X}} \to 2^{\mathbb{F}}$$

defined by $range(\mathbf{a}) = \check{\mathbf{a}}(\mathbf{X})$. Similarly, the concept of domain is viewed as the function

$$domain : \mathbb{F}^{\mathbf{Y}} \to 2^{\mathbf{X}},$$

defined by $domain(\mathbf{a}) = \check{\mathbf{a}}^{-1}(\mathbb{F})$, where **Y** is a subspace of some larger space **X**. In particular, if $\mathbf{a} \in \mathbb{F}^{\mathbf{Y}}$, then $domain(\mathbf{a}) = \mathbf{Y}$.

These mappings can be used to extract point sets and value sets from regions of images of particular interest. For example, the statement

$$s := domain(\mathbf{a}||_{>k}) \tag{4.2.1}$$

yields the *set* of all points (pixel locations) where **a**(**x**) exceeds *k*, namely $s = \{\mathbf{x} \in \mathbf{X} : \mathbf{a}(\mathbf{x}) > k\}$.

Replacing Eqs. 4.2.1 by

$$s := range(\mathbf{a}||_{>k}) \tag{4.2.2}$$

results in a subset of $\mathbb{R}$ instead of **X**.

Restriction, extension, domain, and range are some of the elementary concepts used in the functional specification of image processing techniques. In subsequent sections we shall encounter many others.

4.3 Induced Operations on Images

Manipulating image data using the concepts of restriction, extension, domain, and range does not involve the algebraic structure of the underlying value set; the statement $s := range(\mathbf{a}||_{>k})$ does not reference the arithmetic operations of addition and multiplication associated with $\mathbb{R}$. However, it is the operations associated with the value set $\mathbb{F}$ that are generally used in processing $\mathbb{F}$–valued images. In this section we describe how the operations on $\mathbb{F}$ induce corresponding operations on $\mathbb{F}$–valued images.

Operations on and between $\mathbb{F}$–valued images are the natural induced operations of the algebraic system $\mathbb{F}$. For example, if γ is a binary operation on $\mathbb{F}$, then γ induces a binary operation — again denoted by γ — on $\mathbb{F}^{\mathbf{X}}$ defined as follows:

Let $\mathbf{a}, \mathbf{b} \in \mathbb{F}^{\mathbf{X}}$. Then

$$\mathbf{a}\gamma\mathbf{b} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x})\gamma\mathbf{b}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}. \tag{4.3.1}$$

Induced unary operations are defined in a likewise fashion; any unary operation $f : \mathbb{F} \to \mathbb{F}$ induces a unary operation $f : \mathbb{F}^{\mathbf{X}} \to \mathbb{F}^{\mathbf{X}}$ defined by

$$f(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = f(\mathbf{a}(\mathbf{x})),\ \mathbf{x} \in \mathbf{X}\}. \tag{4.3.2}$$

Note that in this definition we view the composition $f \circ \mathbf{a}$ as a unary operation on $\mathbb{F}^{\mathbf{X}}$ with operand **a**. This subtle distinction has the important consequence that *f* is viewed as a unary operation — namely a function from $\mathbb{F}^{\mathbf{X}}$ to $\mathbb{F}^{\mathbf{X}}$ — and **a** as an *argument* of *f*.

The operations defined by Eqs. 4.3.1 and 4.3.2 are called *induced pixel level operations.* They are also referred to as *grey level based operations* as they operate principally on the pixel or grey values of an image.

In addition to the binary operation defined by Eq. 4.3.1, γ also induces the following scalar operations on images:

For $k \in \mathbb{F}$ and $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$,

$$k\gamma\mathbf{a} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = k\gamma\mathbf{a}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}, \tag{4.3.3}$$

and

$$\mathbf{a}\gamma k = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x})\gamma k,\ \mathbf{x} \in \mathbf{X}\}. \tag{4.3.4}$$

4.3.1 Examples:

(i) Choosing the *l*-ring $(\mathbb{R}, \vee, \wedge, +, \cdot)$ as our value set and applying Eq. 4.3.1, we obtain the following *basic* binary operations on real-valued images:

If $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{\mathbf{X}}$, then

$$\begin{aligned} \mathbf{a} + \mathbf{b} &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) + \mathbf{b}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}, \\ \mathbf{a} \cdot \mathbf{b} &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \cdot \mathbf{b}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}, \\ \mathbf{a} \vee \mathbf{b} &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \vee \mathbf{b}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}, \end{aligned} \tag{4.3.5}$$

and

$$\mathbf{a} \wedge \mathbf{b} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \wedge \mathbf{b}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}. \tag{4.3.6}$$

Obviously, all four operations are commutative and associative.

(ii) For $k \in \mathbb{R}$, we obtain the following scalar multiplication and addition of real-valued images:

$$k \cdot \mathbf{a} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = k \cdot \mathbf{a}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\} \tag{4.3.7}$$

and

$$k + \mathbf{a} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = k + \mathbf{a}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\}. \tag{4.3.8}$$

It follows from the commutativity of real numbers that,

$$k \cdot \mathbf{a} = \mathbf{a} \cdot k \text{ and } k + \mathbf{a} = \mathbf{a} + k.$$

(iii) Following Eq. 4.3.2, we note that any function $f : \mathbb{R} \to \mathbb{R}$ induces a function $f : \mathbb{R}^{\mathbf{X}} \to \mathbb{R}^{\mathbf{X}}$. For example, using the sine function we obtain

$$sin(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = sin(\mathbf{a}(\mathbf{x})),\ \mathbf{x} \in \mathbf{X}\}.$$

As another example, consider the characteristic function

$$\chi_{\geq k}(r) = \begin{cases} 1 & \text{if } r \geq k \\ 0 & \text{otherwise} \end{cases}$$

Then for any $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, $\chi_{\geq k}(\mathbf{a})$ is the boolean (two-valued) image on $\mathbf{X}$ with value 1 at location $\mathbf{x}$ if $\mathbf{a}(\mathbf{x}) \geq k$ and value 0 if $\mathbf{a}(\mathbf{x}) < k$. This type of operation is referred to as *thresholding* and *k* is called the *threshold value* or *threshold level*. Figure **4.3.1** shows an example of a threshold operation. The image $\mathbf{a}$ shown on the left is a 512×512 integer-valued image; i.e., $\mathbf{X} = \mathbb{Z}_{512} \times \mathbb{Z}_{512} \subset \mathbb{Z}^2$. The image on the right is the thresholded image $\chi_{\geq 115}(\mathbf{a})$.

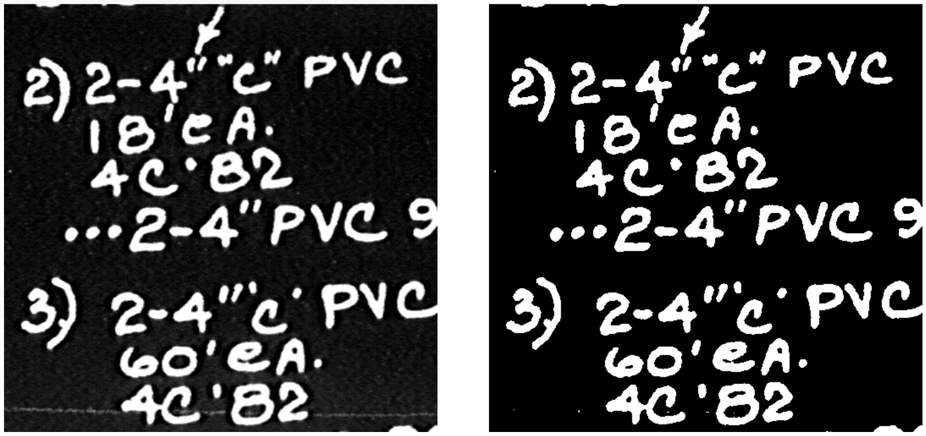

Figure 4.3.1 Application of the characteristic function.

In some threshold operations it is desired to *retain* the original values that pass the threshold. This operation is accomplished by setting

$$\mathbf{c} = \mathbf{a} \cdot \chi_{\geq k}(\mathbf{a})$$

for then $\mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x})$ whenever $\mathbf{a}(\mathbf{x}) \geq k$ and $\mathbf{c}(\mathbf{x}) = 0$ whenever $\mathbf{a}(\mathbf{x}) < k$.

The function $\chi_{\geq k}$ given in the above example is a typical *characteristic function* on the real numbers. In general, given some universal set U and $S \subset U$, then the function $\chi_S : U \to \{0,1\}$ defined by

$$\chi_S(x) = \begin{cases} 1 & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases}$$

is called the *characteristic function* on S. In the above example, $U = \mathbb{R}$ and $S = \{r \in \mathbb{R} : r \geq 115\}$. There is a useful generalization of the concept of a characteristic function. Suppose $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ and $S \in \left(2^{\mathbb{F}}\right)^{\mathbf{X}}$; i.e., $S(\mathbf{x}) \subset \mathbb{F}\ \forall \mathbf{x} \in \mathbf{X}$. We define

$$\chi_S(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) \in S(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\}. \tag{4.3.9}$$

Obviously, if $S : \mathbf{X} \to 2^{\mathbb{F}}$ is a constant function, i.e., S returns the same set $S(\mathbf{x})$ for each $\mathbf{x} \in \mathbf{X}$, then χ_S represents the usual characteristic function. Also note that χ_S returns a boolean-valued image regardless of the type of value set $\mathbb{F}$ used.

4.3.2 **Example:** Pixel level image comparison provides a simple application example of the generalized characteristic function. Given the image $\mathbf{b} \in \mathbb{R}^{\mathbf{X}}$, we define $S_{\leq \mathbf{b}} \in \left(2^{\mathbb{R}}\right)^{\mathbf{X}}$ by $S_{\leq \mathbf{b}}(\mathbf{x}) = \{r \in \mathbb{R} : r \leq \mathbf{b}(\mathbf{x})\}$. The functions $S_{<\mathbf{b}}$, $S_{=\mathbf{b}}$, $S_{\geq \mathbf{b}}$, $S_{>\mathbf{b}}$, $S_{\neq \mathbf{b}}$ are defined analogously. Thus, for example, $S_{=\mathbf{b}} = \{r \in \mathbb{R} : r = \mathbf{b}(\mathbf{x})\}$ and $S_{>\mathbf{b}}(\mathbf{x}) = \{r \in \mathbb{R} : r > \mathbf{b}(\mathbf{x})\}$. Substituting these set functions for S in Eq. 4.3.9 yields

$$\begin{aligned}
\chi_{S_{\leq \mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) \leq \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\} \\
\chi_{S_{<\mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) < \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\} \\
\chi_{S_{=\mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) = \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\} \\
\chi_{S_{\geq \mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) \geq \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\} \\
\chi_{S_{>\mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) > \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\} \\
\chi_{S_{\neq \mathbf{b}}}(\mathbf{a}) &= \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = 1 \text{ if } \mathbf{a}(\mathbf{x}) \neq \mathbf{b}(\mathbf{x}), \text{ otherwise } \mathbf{c}(\mathbf{x}) = 0\}.
\end{aligned} \tag{4.3.10}$$

Here, of course, $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$. In order to reduce and simplify notation, we define

$$\begin{aligned} &\chi_{\leq \mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{\leq \mathbf{b}}}(\mathbf{a}), \ \chi_{<\mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{<\mathbf{b}}}(\mathbf{a}), \ \chi_{\mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{=\mathbf{b}}}(\mathbf{a}), \\ &\chi_{\geq \mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{\geq \mathbf{b}}}(\mathbf{a}), \ \chi_{>\mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{>\mathbf{b}}}(\mathbf{a}), \text{ and } \chi_{\neq \mathbf{b}}(\mathbf{a}) \equiv \chi_{S_{\neq \mathbf{b}}}(\mathbf{a}). \end{aligned} \tag{4.3.11}$$

The characteristic function defined by Eq. 4.3.9 is *not* a naturally induced operation; it is not an operation *on* $\mathbb{F}^{\mathbf{X}}$, but an operation *from* $\mathbb{F}^{\mathbf{X}}$ *to* $\{0,1\}^{\mathbf{X}}$. However, as will be shown in the next section, generalized characteristic functions can nevertheless be derived from the elementary naturally induced operations. Another operation that maps images to a different domain is given by the *global reduce* operation. If γ is an associative and commutative binary operation on $\mathbb{F}$ (i.e., $(\mathbb{F}, \gamma)$ is a commutative semigroup) and $\mathbf{X}$ is finite, say $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, then γ induces a unary operation

$$\Gamma : \mathbb{F}^{\mathbf{X}} \to \mathbb{F}$$

called the *global reduce operation induced by* γ, which is defined as

$$\Gamma \mathbf{a} = \mathop{\Gamma}_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x}) = \mathop{\Gamma}_{k=1}^{n} \mathbf{a}(\mathbf{x}_k) = \mathbf{a}(\mathbf{x}_1) \gamma \mathbf{a}(\mathbf{x}_2) \gamma \cdots \gamma \mathbf{a}(\mathbf{x}_n). \tag{4.3.12}$$

Thus, for example, if $\mathbb{F} = \mathbb{R}$ and γ is the operation of addition ($\gamma = +$), then $\Gamma = \Sigma$ and $\sum \mathbf{a} = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x})$. In all, the lattice $(\mathbb{R}, +, \cdot, \vee, \wedge)$ provides for four basic global reduce operations, namely $\sum \mathbf{a}$, $\prod \mathbf{a}$, $\bigvee \mathbf{a}$, and $\bigwedge \mathbf{a}$.

4.3.3 Examples:

(i) *Area.* If **a** is a boolean image, $\mathbf{a} \in \{0,1\}^{\mathbf{X}}$, where **X** denotes some rectangular subset of $\mathbb{Z}^2$, then $\sum \mathbf{a} = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x})$ counts the number of pixels having value **a(x)** = 1. This count corresponds to a rough estimate of the area occupied by the black objects in the image **a**. Here *black* objects are defined as the connected components (see Section **2.20**) of **X** with the property that **x** is a member of a component $\Leftrightarrow \ \mathbf{a}(\mathbf{x}) = 1$.

(ii) *Image maximum.* If $\mathbb{F} = \mathbb{R}$ and $\gamma = \vee$, then $\Gamma = \bigvee$ and $\bigvee \mathbf{a} = \bigvee_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x})$ denotes the maximum value of the image $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$.

(iii) *Image histogram.* Suppose **X** is a rectangular $m \times n$ array (subset of $\mathbb{Z}^2$), $\mathbf{Y} = \{j \in \mathbb{N} : \ 0 \leq j \leq K\}$ for some fixed integer *K*, and $P = \left\{\mathbf{a} \in \mathbb{N}^{\mathbf{X}} : \ range(\mathbf{a}) \subset \mathbf{Y}\right\}$. Define a parameterized template

$$\mathbf{t} : P \to \left(\mathbb{N}^{\mathbf{Y}}\right)^{\mathbf{X}}$$

by defining for each $\mathbf{a} \in \mathbb{N}^{\mathbf{X}}$ the template **t** with parameter **a**, i.e., $\mathbf{t}(\mathbf{a}) \in \left(\mathbb{N}^{\mathbf{Y}}\right)^{\mathbf{X}}$, by

$$\mathbf{t}(\mathbf{a})_{\mathbf{x}}(j) = \begin{cases} 1 & \text{if } \mathbf{a}(\mathbf{x}) = j \\ 0 & \text{otherwise}. \end{cases}$$

The image $\mathbf{h} \in \mathbb{R}^{\mathbf{Y}}$ obtained from the code

$$\mathbf{h} := \Sigma \mathbf{t}(\mathbf{a})$$

is the histogram of **a**. This follows from the observation that since $\sum \mathbf{t}(\mathbf{a}) = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{t}(\mathbf{a})_{\mathbf{x}}$, $\mathbf{h}(j) = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{t}(\mathbf{a})_{\mathbf{x}}(j) =$ the number of pixels having value *j*.

Figure **4.3.2** provides an example of the process $\mathbf{h} := \Sigma \mathbf{t}(\mathbf{a})$. The left image represents the input image **a** and the right image the graph of the histogram image **h**, resulting from reducing the parameterized template **t**.

If one is interested in the histogram of only one particular value *j*, then it would be more efficient to use the statement

$$n := \sum \chi_{j}(\mathbf{a})$$

since $\sum \chi_{j}(\mathbf{a})$ represents the number of pixels having value j ; i.e., since $\sum \chi_{j}(\mathbf{a}) = \sum_{\mathbf{x} \in \mathbf{X}} \chi_{j}(\mathbf{a}(\mathbf{x}))$ and $\chi_{j}(\mathbf{a}(\mathbf{x})) = 1 \;\Leftrightarrow\; \mathbf{a}(\mathbf{x}) = j$.

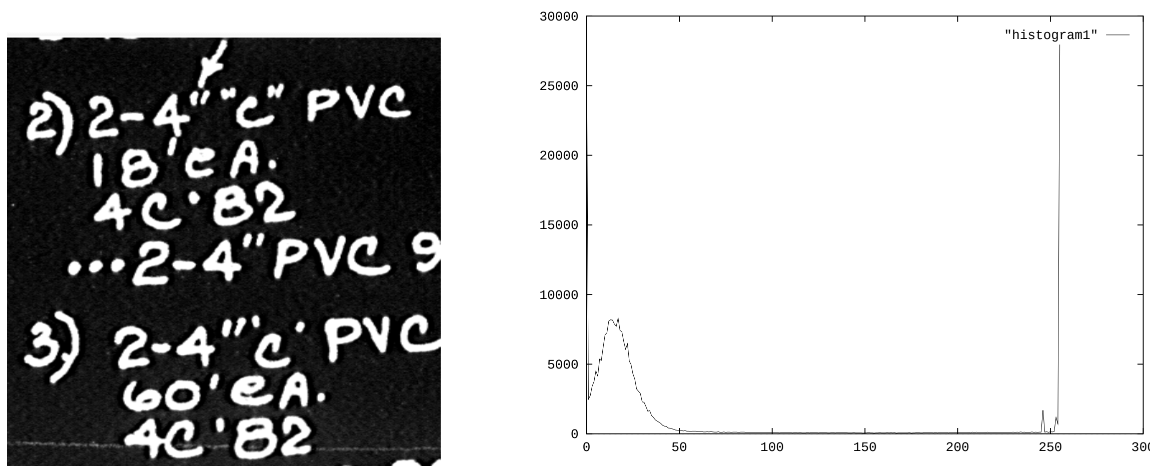

Figure 4.3.2 An image and its histogram.

Note that in the first two examples, the global reduce operation resulted in reducing an image to a single numerical value. In the histogram example, however, we have $\mathbf{t}(\mathbf{a}) \in \mathbb{F}^{\mathbf{X}}$, where $\mathbb{F} = \mathbb{N}^{\mathbf{Y}}$. Thus, by definition $\sum \mathbf{t}(\mathbf{a}) \in \mathbb{F} = \mathbb{N}^{\mathbf{Y}}$ is an integer-valued image on **Y**.

Although in image processing by computer the spatial domain **X** is always finite, the global reduce operation Γ need not be restricted to finite sets. Natural extensions to infinite sets are usually inherent for different binary operations, value sets, and point sets. For example, if **X** is a compact subset of $\mathbb{R}^n$ and $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, then the formulas for area and maximum given in Example **4.3.3** have the form

$$\int \mathbf{a} = \int_{\mathbf{X}} \mathbf{a}(\mathbf{x}) d\mathbf{x} \quad \text{and} \quad \bigvee \mathbf{a} = \bigvee_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x}) = \sup\{\mathbf{a}(\mathbf{x}) : \mathbf{x} \in \mathbf{X}\},$$

respectively. That these are well defined follows from Theorems **2.11.7** and **2.11.8**.

4.4 Properties of $\mathbb{F}^{\mathbf{X}}$

The algebra $\mathbb{F}$ together with the induced algebra $\mathbb{F}^{\mathbf{X}}$ is called an *image algebra*. It follows from the definition of the induced operations (Eqs. 4.3.1 and 4.3.2) that the algebra $\mathbb{F}^{\mathbf{X}}$ inherits most, if not all, of the algebraic properties of the value set $\mathbb{F}$. As an example, consider the set of real–valued images on $\mathbf{X}$. The commutative group $(\mathbb{R}, +)$ induces a commutative group $\left(\mathbb{R}^{\mathbf{X}}, +\right)$. The zero of the group $\mathbb{R}^{\mathbf{X}}$ is the *zero image*, denoted by **0** and defined as $\mathbf{0} = \{(\mathbf{x}, 0) : \mathbf{x} \in \mathbf{X}\}$. Clearly,

$$\mathbf{a} + \mathbf{0} = \mathbf{0} + \mathbf{a} = \mathbf{a}$$

$\forall \mathbf{a} \in \mathbb{R}^{\mathbf{X}}$. Also, each $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$ has an additive inverse $-\mathbf{a}$ defined by $-\mathbf{a} = \{(\mathbf{x}, -\mathbf{a}(\mathbf{x})) : \mathbf{x} \in \mathbf{X}\}$. Obviously,

$$\mathbf{a} + (-\mathbf{a}) = (-\mathbf{a}) + \mathbf{a} = \mathbf{0}.$$

There is nothing special about the group $\mathbb{R}$ in this argument. The same argument can be made for any group $\mathbb{F}$. This establishes the following theorem:

4.4.1 Theorem. *If $(\mathbb{F}, +)$ is a (commutative) group, then $\left(\mathbb{F}^{\mathbf{X}}, +\right)$ is a (commutative) group.*

If, instead of a group, $\mathbb{F}$ is a ring then we also have

4.4.2 Theorem. *If $(\mathbb{F}, +, \cdot)$ is a (commutative) ring, then $\left(\mathbb{F}^{\mathbf{X}}, +, \cdot\right)$ is a (commutative) ring. Furthermore, if $\mathbb{F}$ is a ring with unity, then so is $\mathbb{F}^{\mathbf{X}}$.*

The proof is straightforward; we already know from Theorem **4.4.1** that $\mathbb{F}^{\mathbf{X}}$ is an additive group and it is an elementary exercise to verify that multiplication is associative and distributes over addition. Thus, $\mathbb{F}^{\mathbf{X}}$ is a ring. The *unit image* of $\mathbb{F}^{\mathbf{X}}$ is defined as $\mathbf{1} = \{(\mathbf{x}, 1) : \mathbf{x} \in \mathbf{X}\}$, where 1 denotes the multiplicative identity of $\mathbb{F}$. Obviously,

$$\mathbf{a} \cdot \mathbf{1} = \mathbf{1} \cdot \mathbf{a} = \mathbf{a}$$

$\forall \mathbf{a} \in \mathbb{F}^{\mathbf{X}}$. Thus, if $\mathbb{F}$ is a ring with unity, then so is $\mathbb{F}^{\mathbf{X}}$.

In Section **3.7** (Example **3.7.2**) we noted that $\mathbb{R}^{\mathbf{X}}$ is a vector space over $\mathbb{R}$. Again, there is nothing special about the field $\mathbb{R}$; the same argument can be made for any field $\mathbb{F}$. By Theorem **4.4.1**, $\mathbb{F}^{\mathbf{X}}$ is a group, and using the induced scalar multiplication (Eq. 4.3.3), it is easy to see that the vector space axioms V_1 through V_5 are satisfied. This establishes the following

4.4.3 Theorem. *If $\mathbb{F}$ is a field, then $\mathbb{F}^{\mathbf{X}}$ is a vector space over $\mathbb{F}$. Furthermore, if card*(**X**) = *n* $(n \in \mathbb{N})$, then $\mathbb{F}^{\mathbf{X}}$ is isomorphic to $\mathbb{F}^n$.

The second part of the theorem follows from Example **3.8.3**(i) by defining $\nu \colon \mathbb{F}^{\mathbf{X}} \to \mathbb{F}^{\mathrm{n}}$ by $\nu(\mathbf{a}) = (\mathbf{a}(\mathbf{x}_1), \mathbf{a}(\mathbf{x}_2), \ldots, \mathbf{a}(\mathbf{x}_\mathrm{n}))$, where $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, and then using the exact same argument as in **3.8.3**(i).

It follows from Theorem **4.4.2** that the ring $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ of real-valued images behaves very much like the ring $(\mathbb{R}, +, \cdot)$ of real numbers. In view of the fact that the operations between real-valued images are induced by the operations between real numbers, this should come as no great surprise. Therefore, manipulating real-valued images is analogous to manipulating real numbers, and our familiarity with the real number system provides us with instant familiarity of the induced system $\mathbb{R}^{\mathbf{X}}$. More generally, if we know the system $\mathbb{F}$, then we know the induced system $\mathbb{F}^{\mathbf{X}}$. In image algebra it is always assumed that the algebraic system $\mathbb{F}$ is known and that the algebraic properties of $\mathbb{F}^{\mathbf{X}}$ are then derived from this knowledge. It is important to note, however, that even though the algebraic properties of $\mathbb{F}^{\mathbf{X}}$ are derived from those of $\mathbb{F}$, the overall mathematical structure of $\mathbb{F}^{\mathbf{X}}$ is quite distinct from that of $\mathbb{F}$; elements of $\mathbb{F}^{\mathbf{X}}$ carry spatial information while those of $\mathbb{F}$ generally do not. Furthermore, the induced algebra $\left(\mathbb{F}^{\mathbf{X}}, \gamma\right)$ is structurally not identical to the algebra $(\mathbb{F}, \gamma)$; the induced algebraic structure is usually weaker than the original structure. The succeeding discussion demonstrates this for the *l*-ring of real-valued images.

The operations defined by Eqs. 4.3.5 and 4.3.6 in Example **4.3.1** are the *basic* or *elementary* binary operations on real-valued images. Analogous to the development to the algebra of real numbers, other binary (and unary) operations on real-valued images can be derived either directly or in terms of series expansions from these basic pixel level operations. However, instead of reinventing the wheel, we let the remaining operations on $\mathbb{R}^{\mathbf{X}}$ again be induced by the corresponding operations on $\mathbb{R}$. Two of these operations commonly used in image processing are exponentiation and the computation of logarithms. In particular, if **a** and **b** are real-valued images on **X**, then

$$\mathbf{a}^{\mathbf{b}} = \left\{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x})^{\mathbf{b}(\mathbf{x})},\ \mathbf{x} \in \mathbf{X}\right\}. \tag{4.4.1}$$

Since we are dealing with real-valued images, we follow the rules of arithmetic and restrict this binary operation to those pairs of real-valued images for which $\mathbf{a}(\mathbf{x})^{\mathbf{b}(\mathbf{x})} \in \mathbb{R}\ \forall \mathbf{x} \in \mathbf{X}$. This avoids creation of complex, undefined, and indeterminate pixel values such as $(-1)^{\frac{1}{2}},\ \frac{1}{0^2},\ \text{and}\ 0^0$, respectively. If we consider such subsets as $\mathbb{Z}^{\pm} = \mathbb{Z}^{+} \cup \mathbb{Z}^{-}$ or $\mathbb{R}^{+}$ of $\mathbb{R}$, then exponentiation is defined for all $\mathbf{a}, \mathbf{b} \in (\mathbb{Z}^{\pm})^{\mathbf{X}}$ or $\mathbf{a}, \mathbf{b} \in (\mathbb{R}^{+})^{\mathbf{X}}$.

The inverse of exponentiation is defined in the usual way by taking logarithms. Specifically,

$$log_{\mathbf{b}}\mathbf{a} = \left\{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = log_{\mathbf{b}(\mathbf{x})}\mathbf{a}(\mathbf{x}),\ \mathbf{x} \in \mathbf{X}\right\}. \tag{4.4.2}$$

As for real numbers, $log_{\mathbf{b}}\mathbf{a}$ is defined only for positive images ; i.e., $\mathbf{a}, \mathbf{b} \in (\mathbb{R}^{+})^{\mathbf{X}}$.

If $\mathbf{k} \in \mathbb{F}^{\mathbf{X}}$ is a constant function, then **k** is called a *constant image*. We have already encountered two important real-valued constant images, namely the zero image **0** and the unit image **1**. It follows from Eqs. 4.3.1 and 4.3.3 that

$$0 + \mathbf{a} = \mathbf{0} + \mathbf{a} = \mathbf{a} = 1 \cdot \mathbf{a} = \mathbf{1} \cdot \mathbf{a} \text{ and } 0 \cdot \mathbf{a} = \mathbf{0} \cdot \mathbf{a} = \mathbf{0}.$$

More generally, if $\mathbf{k} \in \mathbb{F}^{\mathbf{X}}$ is the constant image $\mathbf{k}(\mathbf{x}) = k\ \ \forall \mathbf{x} \in \mathbf{X}$, then

$$k\gamma\mathbf{a} = \mathbf{k}\gamma\mathbf{a}.$$

Thus, for $\mathbb{F} = \mathbb{R}$ we have that

$$k + \mathbf{a} = \mathbf{k} + \mathbf{a},\ k \cdot \mathbf{a} = \mathbf{k} \cdot \mathbf{a},\ k \vee \mathbf{a} = \mathbf{k} \vee \mathbf{a}, \text{and } k \wedge \mathbf{a} = \mathbf{k} \wedge \mathbf{a}.$$

The observation that scalar operations on images can be expressed in terms of binary operations between images may be used to define the following unary operations:

$$\mathbf{a}^{k} \equiv \mathbf{a}^{\mathbf{k}},\ k^{\mathbf{a}} \equiv \mathbf{k}^{\mathbf{a}},\ \text{and } log_{k}\mathbf{a} \equiv log_{\mathbf{k}}\mathbf{a}. \tag{4.4.3}$$

These operations are already inherent in Eq. 4.3.2. For example, using the function $f(r) = \frac{1}{r}$ provides for exponentiation by $k = -1$ and yields the inverse $\mathbf{a}^{-1} = f(\mathbf{a})$. In contrast, Eq. 4.4.1 does not follow from Eq. 4.3.2.

Many other unary operations could have been defined directly in terms of elementary binary operations without the use of Eq. 4.3.2. For instance, the absolute value function $abs : \mathbb{R} \to \mathbb{R}$ induces the function $abs : \mathbb{R}^{\mathbf{X}} \to \mathbb{R}^{\mathbf{X}}$ defined by

$$abs(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = |\mathbf{a}(\mathbf{x})|, \ \mathbf{x} \in \mathbf{X}\} = |\mathbf{a}|.$$

However, we could just as easily have defined

$$|\mathbf{a}| \equiv \mathbf{a} \vee (-\mathbf{a}).$$

As alluded to earlier, the generalized characteristic function could have also been defined in terms of elementary binary operations. In order to demonstrate this, we need to take a closer look at the induced structure $\mathbb{R}^{\mathbf{X}}$. Recall that the operation of exponentiation (Eq. 4.4.1) is not defined for all real-valued images. In particular, if $\mathbf{a} \neq \mathbf{0}$ but for some $\mathbf{x} \in \mathbf{X}$ $\mathbf{a}(\mathbf{x}) = 0$, then $\mathbf{a}^{-1}$ does not exist. Thus, in contrast to $(\mathbb{R}, +, \cdot)$, the ring $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ is not a division ring. However, every $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$ has a multiplicative pseudo inverse $\tilde{\mathbf{a}} \in \mathbb{R}^{\mathbf{X}}$ defined by

$$\tilde{\mathbf{a}} = \{(\mathbf{x}, \tilde{\mathbf{a}}(\mathbf{x})) : \tilde{\mathbf{a}}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \text{ if } \mathbf{a}(\mathbf{x}) = 0, \text{ otherwise } \tilde{\mathbf{a}}(\mathbf{x}) = 1/\mathbf{a}(\mathbf{x}), \ \mathbf{x} \in \mathbf{X}\}. \tag{4.4.4}$$

Clearly,

$$\mathbf{a} \cdot \tilde{\mathbf{a}} \cdot \mathbf{a} = \mathbf{a} \text{ and } \tilde{\mathbf{a}} \cdot \mathbf{a} \cdot \tilde{\mathbf{a}} = \tilde{\mathbf{a}}.$$

Thus $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ is a von Neumann ring.

Although, in general, $\tilde{\mathbf{a}} \cdot \mathbf{a}$ need not equal the unit image $\mathbf{1}$, $\tilde{\mathbf{a}} \cdot \mathbf{a}$ is always a boolean image. Keeping these observations in mind, it is easy to see that if $\mathbf{c} = (\mathbf{a} - \mathbf{b}) \vee \mathbf{0}$, then the boolean image $\tilde{\mathbf{c}} \cdot \mathbf{c}$ satisfies the equation

$$\chi_{>\mathbf{b}}(\mathbf{a}) = \tilde{\mathbf{c}} \cdot \mathbf{c}\,.$$

Thus, the characteristic function $\chi_{>\mathbf{b}}$ can be readily expressed in terms of elementary binary operations. The function $\chi_{<\mathbf{b}}$ can be defined in a similar manner and the remaining characteristic functions are then derived from $\chi_{>\mathbf{b}}$ and $\chi_{<\mathbf{b}}$ by use of Boolean complementation and multiplication. Defining

$$\mathbf{a}^c = \mathbf{1} - \tilde{\mathbf{a}} \cdot \mathbf{a}\,, \tag{4.4.5}$$

we obtain

$$\chi_{\leq\mathbf{b}}(\mathbf{a}) = [\chi_{>\mathbf{b}}(\mathbf{a})]^c, \ \ \chi_{\geq\mathbf{b}}(\mathbf{a}) = [\chi_{<\mathbf{b}}(\mathbf{a})]^c,$$
$$\chi_{\mathbf{b}}(\mathbf{a}) = \chi_{\leq\mathbf{b}}(\mathbf{a}) \cdot \chi_{\geq\mathbf{b}}(\mathbf{a}), \text{ and } \chi_{\neq\mathbf{b}}(\mathbf{a}) = [\chi_{\mathbf{b}}(\mathbf{a})]^c.$$

This verifies our earlier claim.

Suppose $\mathbf{a} \neq \mathbf{0}$ but for some $\mathbf{x} \in \mathbf{X}$ $\mathbf{a}(\mathbf{x}) = 0$. Setting $\mathbf{b} = \mathbf{1} - \tilde{\mathbf{a}} \cdot \mathbf{a}$, we have $\mathbf{a} \cdot \mathbf{b} = \mathbf{0}$ while neither $\mathbf{a} \neq \mathbf{0}$ nor $\mathbf{b} \neq \mathbf{0}$. Therefore $\mathbf{a}$ and $\mathbf{b}$ are divisors of zero, which says that $\mathbb{R}^{\mathbf{X}}$ is not an integral domain. Of course, we have already noted that $\mathbb{R}^{\mathbf{X}}$ is not a division ring. The obvious question then is as to how closely the induced structure $\mathbb{F}^{\mathbf{X}}$ resembles the base structure $\mathbb{F}$. In particular, if $\mathbb{F}$ is a division ring or field, how close is $\mathbb{F}^{\mathbf{X}}$ to being a division ring or field? From our above discussion, it is easily inferred that the following theorem is the best we can hope for.

4.4.4 Theorem. *If* $(\mathbb{F}, +, \cdot)$ *is a (commutative) division ring, then* $\left(\mathbb{F}^{\mathbf{X}}, +, \cdot\right)$ *is a (commutative) von Neumann ring.*

The proof is elementary. According to Theorem **4.4.2**, $\mathbb{F}^{\mathbf{X}}$ is a ring and if $\mathbb{F}$ is a commutative ring, then so is $\mathbb{F}^{\mathbf{X}}$. Pseudo inverses are defined in the exact same way as pseudo inverses for elements of $\mathbb{R}^{\mathbf{X}}$ (Eq. 4.4.4) by replacing the number 0 with the additive identity of $\mathbb{F}$.

Since for any image $\mathbf{a} \neq \mathbf{0}$ with the property that for some $\mathbf{x} \in \mathbf{X}$ $\mathbf{a}(\mathbf{x}) = 0$, we are unable to obtain an inverse, $\mathbb{F}^{\mathbf{X}}$ cannot be a division ring. Hence, $\mathbb{F}$ and $\mathbb{F}^{\mathbf{X}}$ can structurally never be the same — except in the trivial case *card*($\mathbf{X}$) = 1. However, since $\mathbb{F}^{\mathbf{X}}$ is a von Neumann ring, it is *almost* a division ring in that every non-zero element has *almost* a multiplicative inverse, namely a pseudo inverse.

If instead of using group, ring, or vector space operations, we consider lattice operations we encounter similar observations. The lattice $\left(\mathbb{R}^{\mathbf{X}}, \vee, \wedge\right)$ behaves very much like the lattice $(\mathbb{R}, \vee, \wedge)$. For example, the equality

$$\mathbf{a} \wedge \mathbf{b} = -(-\mathbf{a} \vee -\mathbf{b})$$

holds in both lattices. Inequalities for images can be defined in terms of the semigroup operation $\vee$ (or $\wedge$) by

$$\mathbf{a} \leq \mathbf{b} \Leftrightarrow \mathbf{a} \vee \mathbf{b} = \mathbf{b} \quad (\text{or } \mathbf{a} \leq \mathbf{b} \Leftrightarrow \mathbf{a} \wedge \mathbf{b} = \mathbf{a}).$$

However, if $\mathbf{X}$ contains more than one point, then it is possible to have two real-valued images $\mathbf{a}$ and $\mathbf{b}$ with $\mathbf{a} \neq \mathbf{a} \vee \mathbf{b}$ *and* $\mathbf{b} \neq \mathbf{a} \vee \mathbf{b}$. Thus we have images $\mathbf{a}$ and $\mathbf{b}$ such that neither statement $\mathbf{a} \leq \mathbf{b}$ nor $\mathbf{b} \leq \mathbf{a}$ is true. Hence the lattice $\mathbb{R}^{\mathbf{X}}$ is only partially ordered and can, therefore, structurally not be identical to the lattice $\mathbb{R}$. This corroborates our earlier claim that the induced structure $\mathbb{F}^{\mathbf{X}}$ is somewhat weaker than the base structure $\mathbb{F}$.

We now address the question of functionality of the induced operations for expressing image processing algorithms. The procedure of combining a finite number of images using a finite number of elementary induced image operations results in some new image or other value, such as a scalar. The result of writing down such a procedure is called an *image algebra expression*. A transform $\tau : \mathbb{F}^{\mathbf{X}} \to \mathbb{F}^{\mathbf{Y}}$ defined in terms of an image algebra expression is called an *image algebra transform*. Thus $\tau(\mathbf{a})$ is an image algebra expression for each $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$. For example, the transforms $\tau, \rho : \mathbb{R}^{\mathbf{X}} \to \mathbb{R}^{\mathbf{X}}$ defined by $\tau(\mathbf{a}) = \mathbf{a} \vee (-\mathbf{a})$ and $\rho(\mathbf{a}) = \mathbf{a} \cdot \mathbf{b} + \mathbf{a} \cdot \mathbf{a}$ are both image algebra transforms. In the second transform we assume that $\mathbf{b}$ is some fixed given image, and $\mathbf{a}$ the variable. We conclude this section by showing that any algorithm — or, in fact, any transformation — that transforms a digital image into a digital, floating point, or any other type of real-valued image, can be realized by an image algebra expression with operations consisting only of the induced elementary operations of addition and multiplication. The main ingredient in proving this claim is the following interpolation theorem.

4.4.5 Theorem. *(Lagrange Interpolation). For each integer* $j = 0, 1, \ldots, m$ *there exists a polynomial* $h_j(x)$ *such that*

$$h_j(i) = \begin{cases} 1 & if\ \ i = j \\ 0 & if\ \ i \neq j\,. \end{cases}$$

Proof: Let $L_j(x) = \prod_{i=0, i\neq j}^{m} (x-i)$; i.e.,

$$L_j(x) = x(x-1)(x-2)\cdots(x-(j-1))(x-(j+1))\cdots(x-m).$$

Then $L_j(i) = 0$ if $i \neq j$, and $L_j(i) \neq 0$ whenever $i = j$. Now define

$$h_j(x) = \frac{1}{L_j(j)} L_j(x).$$

Q.E.D.

For the remainder of this section we suppose that **X** is a finite point set, say $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_m\}$. Now if $p \in \mathbb{R}[x_0, x_1, \ldots, x_m]$ is a polynomial in $m+1$ variables, then *p* gives rise to a transformation $\bar{p} : \mathbb{R}^{\mathbf{X}} \to \mathbb{R}^{\mathbf{X}}$ defined by

$$\bar{p}(\mathbf{a}) = p(\mathbf{a}(\mathbf{x}_0)\cdot\mathbf{1}, \mathbf{a}(\mathbf{x_1})\cdot\mathbf{1}, \ldots, \mathbf{a}(\mathbf{x}_m)\cdot\mathbf{1}),$$

where addition and multiplication in $p(x_0, x_1, \ldots, x_m)$ are now interpreted as the induced addition and multiplication in $\mathbb{R}^{\mathbf{X}}$. Thus, $\bar{p}(\mathbf{a})$ is an expression in the image algebra $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$.

Defining for each integer *j* the constant image $\mathbf{j} = j \cdot \mathbf{1}$, we obtain the following interpretation of Theorem **4.4.5**:

4.4.6 **Corollary.** *(Image Interpolation). For each constant image $\mathbf{j} = \mathbf{0}, \mathbf{1}, \ldots, \mathbf{m}$, there exists an image algebra expression $\bar{h}_{\mathbf{j}}(\mathbf{a})$ in $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ such that*

$$\bar{h}_{\mathbf{j}}(\mathbf{i}) = \begin{cases} \mathbf{1} & \text{if } \mathbf{i} = \mathbf{j} \\ \mathbf{0} & \text{if } \mathbf{i} \neq \mathbf{j} \end{cases}$$

Proof: Defining $\bar{L}_{\mathbf{j}}(\mathbf{a}) = \prod_{\mathbf{i}=\mathbf{0}, \mathbf{i}\neq\mathbf{j}}^{\mathbf{m}} (\mathbf{a} - \mathbf{i})$, we note that the image $\mathbf{c}_j = \bar{L}_{\mathbf{j}}(\mathbf{j})$ has the property that $\mathbf{c}_j(\mathbf{x}) \neq 0 \ \ \forall \mathbf{x} \in \mathbf{X}$. Thus $\left(\bar{L}_{\mathbf{j}}(\mathbf{j})\right)^{-1} = \frac{\mathbf{1}}{\bar{L}_{\mathbf{j}}(\mathbf{j})}$ exists for $\mathbf{j} = \mathbf{0}, \mathbf{1}, \ldots, \mathbf{m}$. It is also obvious that $\bar{L}_{\mathbf{j}}(\mathbf{i}) = \mathbf{0}$ whenever $\mathbf{i} \neq \mathbf{j}$. Thus, the expression $\bar{h}_{\mathbf{j}}(\mathbf{a}) = \left[\bar{L}_{\mathbf{j}}(\mathbf{j})\right]^{-1} \cdot \left[\bar{L}_{\mathbf{j}}(\mathbf{a})\right]$ has the desired properties.

Q.E.D.

We conclude by proving that the algebra $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ is sufficient for expressing all digital image to real-valued image transformations.

4.4.7 **Theorem.** *Suppose $\varphi : (\mathbb{Z}_{2^n})^{\mathbf{X}} \to \mathbb{R}^{\mathbf{X}}$ is any transformation. Then there exists an expression τ in the image algebra $\left(\mathbb{R}^{\mathbf{X}}, +, \cdot\right)$ such that $\tau(\mathbf{a}) = \varphi(\mathbf{a}) \ \forall \mathbf{a} \in (\mathbb{Z}_{2^n})^{\mathbf{X}}$.*

Proof: Since $card(\mathbb{Z}_{2^n})^{\mathbf{X}} = 2^{n(m+1)}$, we may assume without loss of generality that $(\mathbb{Z}_{2^n})^{\mathbf{X}} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_\zeta\}$, where $\zeta = 2^{n(m+1)}$.
For each $k = 1, 2, \ldots, \zeta$, let $p_k \in \mathbb{R}[x_0, x_1, \ldots, x_m]$ be defined as follows:

Let

$$k_0 = \mathbf{a}_k(\mathbf{x}_0),\ k_1 = \mathbf{a}_k(\mathbf{x}_1),\ \ldots\ ,\ k_m = \mathbf{a}_k(\mathbf{x}_m)$$

and set $\mathbf{k}_j = k_j \cdot \mathbf{1}$, where $j = 0, 1, \ldots, m$. Now define

$$p_k(x_0\,, x_1\,, \ldots, x_m) = h_{k_0}(x_0) h_{k_1}(x_1) \cdots h_{k_m}(x_m),$$

where $h_{k_i}(x)$ is the polynomial in Theorem **4.4.5**.

Obviously,

$$\begin{aligned}\bar{p}_k(\mathbf{a}_k) &= p_k(\mathbf{a}_k(\mathbf{x}_0)\cdot\mathbf{1}, \mathbf{a}_k(\mathbf{x}_1)\cdot\mathbf{1}, \ldots, \mathbf{a}_k(\mathbf{x}_m)\cdot\mathbf{1}) \\ &= p_k(\mathbf{k}_0, \mathbf{k}_1, \ldots, \mathbf{k}_m) \\ &= \bar{h}_{\mathbf{k}_0}(\mathbf{k}_0)\cdot\bar{h}_{\mathbf{k}_1}(\mathbf{k}_1)\cdots\bar{h}_{\mathbf{k}_m}(\mathbf{k}_m) = \mathbf{1}\,.\end{aligned}$$

Now consider $\bar{p}_k(\mathbf{a}_j)$, where $j \neq k$. Then $\mathbf{a}_j \neq \mathbf{a}_k$ and , therefore, $\mathbf{a}_j(\mathbf{x}_i) \neq \mathbf{a}_k(\mathbf{x}_i)$ for at least one point $\mathbf{x}_i \in \mathbf{X}$. Thus, $\mathbf{j}_i \neq \mathbf{k}_i$ and, hence, $\bar{h}_{\mathbf{k}_i}(\mathbf{j}_i) = \mathbf{0}$.

Therefore,

$$\bar{p}_k(\mathbf{a}_j) = \begin{cases}\mathbf{1} & \text{if } j = k \\ \mathbf{0} & \text{if } j \neq k\end{cases} \tag{I}$$

For each $k = 1, 2, \ldots, \zeta$, let $\mathbf{b}_k = \varphi(\mathbf{a}_k)$ and define τ by

$$\tau(\mathbf{a}) = \bar{p}_1(\mathbf{a})\cdot\mathbf{b}_1 + \bar{p}_2(\mathbf{a})\cdot\mathbf{b}_2 + \cdots + \bar{p}_\zeta(\mathbf{a})\cdot\mathbf{b}_\zeta\,.$$

It now follows from Eq. (I) that $\tau(\mathbf{a}_k) = \varphi(\mathbf{a}_k)$ and, therefore, $\tau(\mathbf{a}) = \varphi(\mathbf{a})\ \forall\mathbf{a} \in (\mathbb{Z}_{2^n})^{\mathbf{X}}$.

Q.E.D.

The theorem is primarily of theoretical interest as it provides no practical method for determining τ from φ. Of course, the hypothesis of the theorem is general enough to cover all image-to–image transformations that occur in practice as only finitely many grey values can be represented.

4.4.8 **Corollary.** *Suppose* $\mathbf{X}$ *and* $\mathbf{Y}$ *are two finite point sets. If* $\varphi : (\mathbb{Z}_{2^n})^{\mathbf{X}} \to \mathbb{R}^{\mathbf{Y}}$ *is any transformation, then there exists an expression* τ *in the image algebra* $\left(\mathbb{R}, \mathbb{R}^{\mathbf{X}}, +, \cdot, \Sigma\right)$ *such that* $\tau(\mathbf{a}) = \varphi(\mathbf{a})\ \forall\mathbf{a} \in (\mathbb{Z}_{2^n})^{\mathbf{X}}$.

Proof: Let $p_1, p_2, \ldots, p_\zeta$ be polynomials such that

$$\bar{p}_k(\mathbf{a}_j) = \begin{cases}\mathbf{1} & \text{if } j = k \\ \mathbf{0} & \text{if } j \neq k\end{cases}$$

Again, let $\mathbf{b}_k = \varphi(\mathbf{a}_k)$, where $\mathbf{b}_k$ is now an image on $\mathbf{Y}$. Finally, define τ by

$$\tau(\mathbf{a}) = \left[\sum\left(\frac{1}{m+1}\cdot\bar{p}_1(\mathbf{a})\right)\right]\cdot\mathbf{b}_1 + \cdots + \left[\sum\left(\frac{1}{m+1}\cdot\bar{p}_\zeta(\mathbf{a})\right)\right]\cdot\mathbf{b}_\zeta\,.$$

Then $\tau(\mathbf{a}_k) = \left[\sum\left(\frac{1}{m+1}\cdot\bar{p}_k(\mathbf{a}_k)\right)\right]\cdot\mathbf{b}_k = \mathbf{b}_k = \varphi(\mathbf{a}_k)$, and, therefore, $\tau(\mathbf{a}) = \varphi(\mathbf{a})\ \forall\mathbf{a} \in (\mathbb{Z}_{2^n})^{\mathbf{X}}$.

Q.E.D.

The corollary is useful when considering transformations of images to images defined on different spatial domains.

4.5 Spatial Operations

Examples of spatial based image transformations are affine and perspective transforms. These type of transforms are commonly used in image registration and rectification. The need for registration and rectification arises in digital image processing when the positions of some or all image pixel locations are significantly displaced from their true locations on a uniformly scaled grid. For example, objects in magnetic resonance imagery (MRI) are displaced because of the warping effects of the magnetic field and raster-scanned satellite images of the earth exhibit the phenomenon that adjacent scan lines are offset slightly with respect to one another because the earth rotates as successive lines of an image are recorded. Since the induced image operations represented by Eqs. 4.3.1 through 4.3.4 are grey level based, the induced algebraic structure does not provide for either intuitive or simple expressions of these types of transforms. The purpose of this section is to resolve this difficulty by providing a seamless extension of the image algebra defined in the previous sections.

Viewing an image $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ as a function from $\mathbf{X}$ to $\mathbb{F}$, we note that the induced unary operation 4.3.2 is actually the composition $f \circ \mathbf{a} = f(\mathbf{a})$ of the image function $\mathbf{a}$ with the function f from $\mathbb{F}$ to $\mathbb{F}$ as shown in Fig. **4.5.1**. Taking the same viewpoint, but using a function f between spatial domains instead,

$$
\begin{tikzcd}
\mathbf{X} \arrow[d, "\mathbf{a}"'] \arrow[dr, "f\circ\mathbf{a}"] & \\
\mathbb{F} \arrow[r, "f"] & \mathbb{F}
\end{tikzcd}
$$

Figure 4.5.1 The induced grey level transform $f \circ \mathbf{a} = f(\mathbf{a})$.

provides a scheme for realizing naturally induced operations for spatial manipulation of image data. In particular, if $f : \mathbf{Y} \to \mathbf{X}$ and $\mathbf{a} \in \mathbb{F}^{\mathbf{X}}$, then we define the induced image $\mathbf{a} \circ f \in \mathbb{F}^{\mathbf{Y}}$ by

$$\mathbf{a} \circ f = \{(\mathbf{y}, \mathbf{a}(f(\mathbf{y}))) : \mathbf{y} \in \mathbf{Y}\} . \tag{4.5.1}$$

Thus, the operation defined by Eq. 4.5.1 transforms an $\mathbb{F}$–valued image defined over the space $\mathbf{X}$ into an $\mathbb{F}$–valued image defined over the space $\mathbf{Y}$. Figure **4.5.2** provides a visual interpretation of this composition operation.

$$
\begin{tikzcd}
\mathbf{X} \arrow[d, "\mathbf{a}"'] & \mathbf{Y} \arrow[l, "f"'] \arrow[dl, "\mathbf{a} \circ f"] \\
\mathbb{F} &
\end{tikzcd}
$$

Figure 4.5.2 The spatial transform $\mathbf{a} \circ f$.

In Section **4.8**, it will become evident that the induced image can just as easily be obtained by use of an image–template operation. However, in various cases Eq. 4.5.1 provides for a more translucent expression and computationally more efficient method than an image-template product.

4.5.1 Examples:

(i) *One sided reflection.* Suppose $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, where $\mathbf{X} \subset \mathbb{Z}^2$ is a rectangular $m \times n$ array. If $1 \leq k \leq \frac{m}{2}$ and $f : \mathbf{X} \to \mathbf{X}$ is defined as

$$
f(x, y) = \begin{cases} (x, y) & \text{if } k \leq x \\ (2k - x, y) & \text{if } x < k \end{cases},
$$

then $\mathbf{a} \circ f$ is a one sided reflection of **a** across the line $x = k$. Figure **4.5.3** illustrates such a reflection on a 512×512 image across the line $k = 180$.

Figure 4.5.3 One sided reflection across a line.

(ii) *Magnification by replication.* For $r \in \mathbb{R}$, let $\lceil r \rceil$ denote the smallest integer greater or equal to *r*; i.e., $\lceil r \rceil - 1 < r \leq \lceil r \rceil$, where $\lceil r \rceil \in \mathbb{Z}$. For a given pair of real numbers $\mathbf{y} = (y_1, y_2)$, let $\lceil \mathbf{y} \rceil = (\lceil y_1 \rceil, \lceil y_2 \rceil)$. Now let **a** and **X** be as in the previous example, and let **Y** be a $km \times kn$ array, where *k* is some positive integer. Define $f : \mathbf{Y} \to \mathbf{X}$ by $f(\mathbf{y}) = \left\lceil \frac{1}{k}\mathbf{y} \right\rceil$; i.e., $f(y_1, y_2) = (x_1, x_2)$, where $x_i = \left\lceil \frac{1}{k} y_i \right\rceil$. Then $\mathbf{a} \circ f$ represents the magnification of **a** by a factor of *k*. Figure **4.5.4** represents an example of a magnification by a factor of $k = 2$. Here each pixel was replicated four times, two times in the *x* direction and two times in the *y* direction.

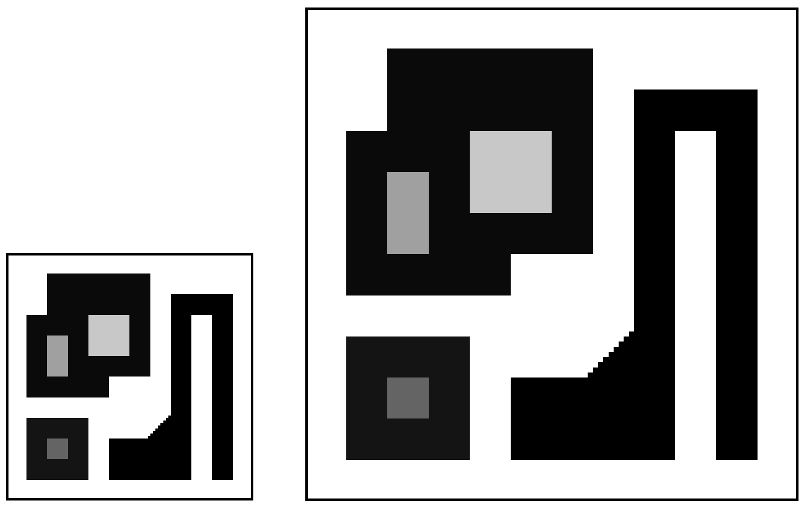

Figure 4.5.4 Magnification by replication.

Note that in both, image reflection and image magnification, the function *f* can be viewed as a function on $\mathbb{Z}^2$ into $\mathbb{Z}^2$; that is, as a function preserving integral coordinates. The general definition for geometric operations in the plane $\mathbb{R}^2$ is in terms of a function $f : \mathbb{R}^2 \to \mathbb{R}^2$ with

$$f(x, y) = (f_1(x, y), f_2(x, y)), \tag{4.5.2}$$

resulting in the expression

$$\mathbf{b}(x, y) = \mathbf{a}(x', y') = \mathbf{a}(f_1(x, y), f_2(x, y)), \tag{4.5.3}$$

where **a** denotes the input image, **b** the output image, $x' = f_1(x, y)$, and $y' = f_2(x, y)$. In digital image processing, the grey level values of the input image are defined only at integral values of x' and y'. However, restricting *f* to some rectangular subset **Y** of $\mathbb{Z}^2 \subset \mathbb{R}^2$, it is obvious that for most functions *f*, $range(f) \not\subset \mathbb{Z}^2$. This situation is illustrated in Figure **4.5.5**, where an output location is mapped to a position between four input pixels. For example, if *f* denotes the rotation about the origin, then the coordinates

$$x' = f_1(x, y) = x cos\theta - y sin\theta \ \text{ and } \ y' = f_2(x, y) = x sin\theta + y cos\theta \tag{4.5.4}$$

do not, in general, correspond to integral coordinates but will lie between an adjacent integer pair. Thus, the formulation given by Eq. 4.5.1 can usually not be used directly in digital image processing. One way of solving this problem is by simply redefining *f* as $\hat{f}(x, y) \equiv [f(x, y)]$. That is,

$$\hat{f}(x, y) = ([x'], [y']) \Leftrightarrow f(x, y) = (x', y'),$$

where $[r]$ denotes the rounding of r to the nearest integer. Then the image $\mathbf{b} = \mathbf{a} \circ \hat{f}$ is said to be obtained from **a** by using the *nearest neighbor*, or *zero-order interpolation*. In this case, the grey level of the output pixel is taken to be that of the input pixel nearest to the position to which it maps. This is computationally efficient and, in many cases, produces acceptable results. On the negative side, this simple interpolation scheme can introduce artifacts in images whose grey levels change significantly over one unit of pixel spacing. Figure **4.5.6** shows an example of rotating an image using nearest neighbor interpolation. The results show a sawtooth effect at the edges.

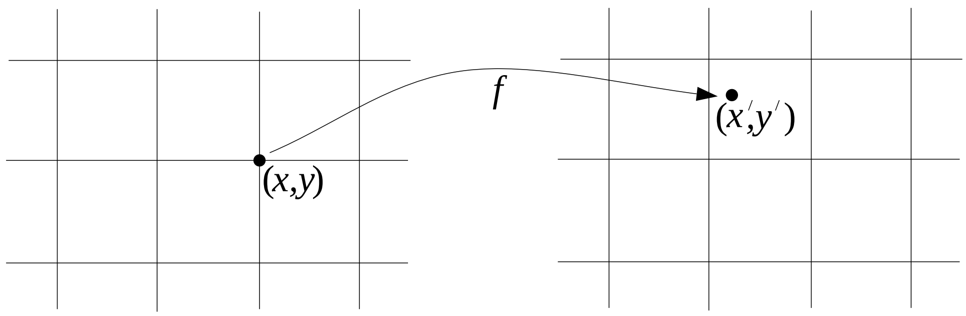

Figure 4.5.5 Mapping of integral to non-integral coordinates.

As mentioned previously, the output pixels map to non-integral positions of the input image array, generally falling between four pixels, and simple nearest neighbor computation can produce undesirable artifacts. Thus, higher order interpolation schemes are often necessary to determine the pixel values of the output image. The method of choice is *first-order* or *bilinear interpolation.* First-order interpolation produces more desirable results with only a modest increase in computational complexity.

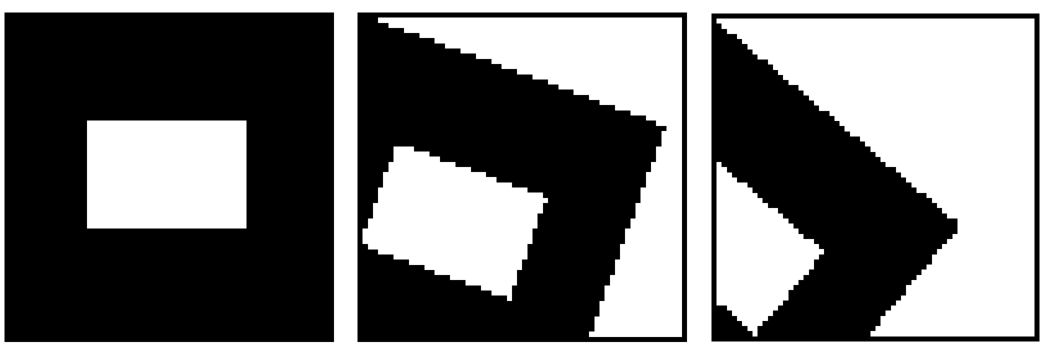

Figure 4.5.6 Rotation using nearest neighbor interpolation.

Suppose

$$\mathbf{X} = \{(x_1, x_2) \,:\, 1 \le x_1 \le m,\ 1 \le x_2 \le n,\ \ (x_1, x_2) \in \mathbb{Z}^2\}$$

and

$$\mathbf{X}' = \{(x_1', x_2') \,:\, 1 \le x_1' \le m,\ 1 \le x_2' \le n,\ \ (x_1', x_2') \in \mathbb{R}^2\}.$$

Given $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, we extend **a** to a function $\mathbf{a}' \in (\mathbb{R})^{\mathbb{R}^2}$ as follows:
First set both $\mathbf{a}(x_1, x_2) = 0$ and $\mathbf{a}'(x_1, x_2) = 0$ whenever $(x_1, x_2) \in \mathbb{R}^2 \backslash \mathbf{X}'$. For $(x_1', x_2') \in \mathbf{X}'$, set

$$\begin{aligned} \mathbf{a}'(x_1', x_2') &= \mathbf{a}(x_1, x_2) + [\mathbf{a}(x_1 + 1, x_2) - \mathbf{a}(x_1, x_2)](x_1' - x_1) \\ &+ [\mathbf{a}(x_1, x_2 + 1) - \mathbf{a}(x_1, x_2)](x_2' - x_2) \\ &+ [\mathbf{a}(x_1 + 1, x_2 + 1) + \mathbf{a}(x_1, x_2) - \mathbf{a}(x_1, x_2 + 1) - \mathbf{a}(x_1 + 1, x_2)](x_1' - x_1)(x_2' - x_2), \end{aligned} \tag{4.5.5}$$

where $x_i = \lfloor x_i' \rfloor$ and $\lfloor x_i' \rfloor$ denotes the largest integer with the property $\lfloor x_i' \rfloor \leq x_i'$. Note that $x_i \leq x_i' \leq x_i + 1$, and $\mathbf{a}'(x_1', x_2') = \mathbf{a}(x_1, x_2)$ whenever $x_1 = x_1'$ and $x_2 = x_2'$. Thus, $\mathbf{a}'$ is a continuous extension of **a** to $\mathbf{X}'$ with the four corner values of **a** agreeing with those of $\mathbf{a}'$ (Fig. **4.5.7**).

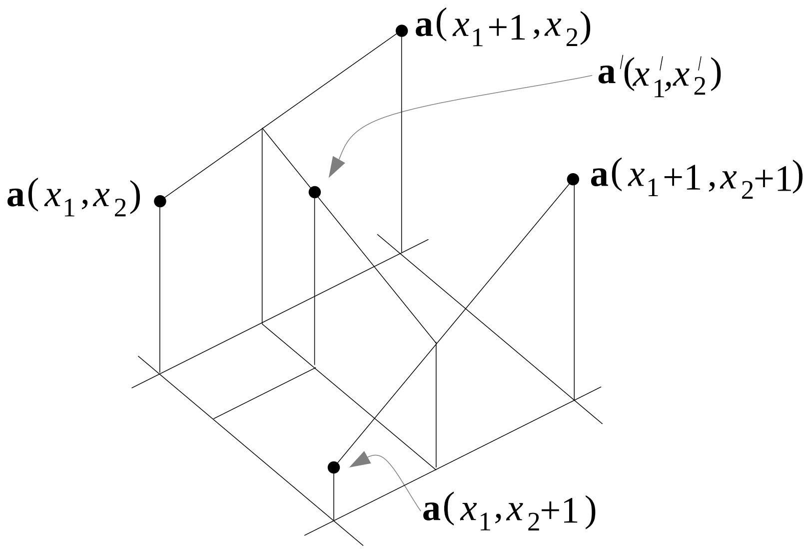

Figure 4.5.7 The bilinear interpolation of **a**.

Now given a transformation $f : \mathbb{R}^2 \to \mathbb{R}^2$ and an output array **Y**, we view f as restricted to $f : \mathbf{Y} \to \mathbb{R}^2$ and set

$$\mathbf{b} = \mathbf{a}' \circ f \tag{4.5.6}$$

in order to obtain the desired spatial transformation of **a**. In particular, the values of x_i' for $i = 1, 2$ in Eq. 4.5.5 are now replaced by the coordinate functions $f_i(y_1, y_2)$ of f. Figure **4.5.8** illustrates a rotation using first order interpolation. Note that in comparison to the rotation using zero-order interpolation (Fig. **4.5.6**), the boundary of the small interior rectangle has a smoother appearance; the sawtooth effect has disappeared. The *outer* boundary of the rotated image retains the sawtooth appearance since no interpolation occurs on points $(x_1', x_2') \in \mathbb{R}^2 \setminus \mathbf{X}'$.

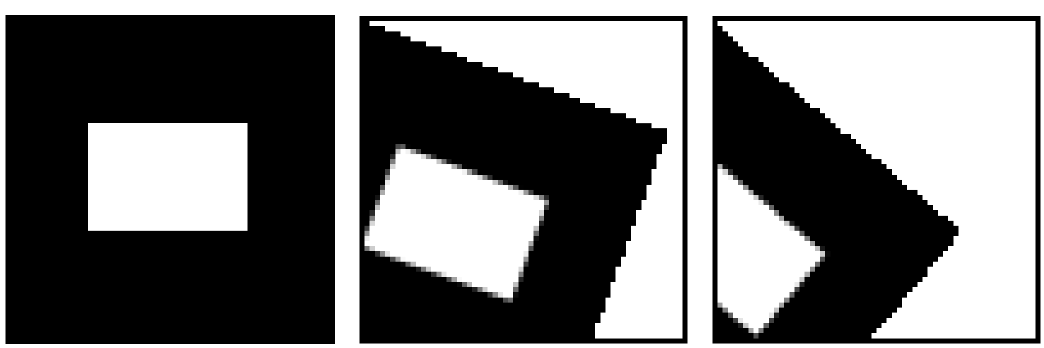

Figure 4.5.8 Rotation using first-order interpolation.

The computational complexity of the bilinear interpolation (Eq. 4.5.5) can be improved if we first interpolate along one direction twice and then along the other direction once. Specifically, Eq. 4.5.5 can be decomposed as follows. For a given point $(x_1', x_2') \in \mathbf{X}'$ compute

$$\mathbf{a}_1'(x_1', x_2) = \mathbf{a}(x_1, x_2) + [\mathbf{a}(x_1+1, x_2) - \mathbf{a}(x_1, x_2)](x_1' - x_1)$$

along the line segment with endpoints (x_1, x_2) and $(x_1 + 1, x_2)$, and

$$\mathbf{a}_2'(x_1', x_2+1) = \mathbf{a}(x_1, x_2+1) + [\mathbf{a}(x_1+1, x_2+1) - \mathbf{a}(x_1, x_2+1)](x_1' - x_1)$$

along the line segment with endpoints (x_1, x_2+1) and (x_1+1, x_2+1) . Then set

$$\mathbf{a}'(x_1', x_2') = \mathbf{a}_1'(x_1', x_2) + \left[\mathbf{a}_2'(x_1', x_2+1) - \mathbf{a}_1'(x_1', x_2)\right](x_2' - x_2).$$

This reduces the four multiplications and eight additions or subtractions inherent in Eq. 4.5.5 to only three multiplications and six additions/subtractions.

Although Eq. 4.5.6 represents a functional specification of a spatial image transformation, it is somewhat deceiving; the image $\mathbf{a}'$ in the equation was derived using typical algorithmic notation (Eq. 4.5.5). To obtain a functional specification for the interpolated image $\mathbf{a}'$ we can specify its values using three spatial transformations f_0, f_1, and f_2, mapping $\mathbf{X}' \rightarrow \mathbf{X}$, defined by

$$f_0(x_1', x_2') = (\lfloor x_1' \rfloor, \lfloor x_2' \rfloor)$$

$$f_1(x_1', x_2') = \begin{cases} (\lfloor x_1' \rfloor + 1, \lfloor x_2' \rfloor) & \text{if } \lfloor x_1' \rfloor < m \\ (\lfloor x_1' \rfloor, \lfloor x_2' \rfloor) & \text{otherwise} \end{cases}$$

and

$$f_2(x_1', x_2') = \begin{cases} (\lfloor x_1' \rfloor, \lfloor x_2' \rfloor + 1) & \text{if } \lfloor x_2' \rfloor < n \\ (\lfloor x_1' \rfloor, \lfloor x_2' \rfloor) & \text{otherwise} \end{cases},$$

and two real-valued images functions $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^{\mathbf{X}'}$ defined by

$$\mathbf{w}_1(x_1', x_2') = x_1' - \lfloor x_1' \rfloor$$

and

$$\mathbf{w}_2(x_1', x_2') = x_2' - \lfloor x_2' \rfloor.$$

We now define

$$\begin{aligned} \mathbf{a}' = {} & \mathbf{a} \circ f_0 + (\mathbf{a} \circ f_1 - \mathbf{a} \circ f_0) \cdot \mathbf{w}_1 + (\mathbf{a} \circ f_2 - \mathbf{a} \circ f_0) \cdot \mathbf{w}_2 \\ & + (\mathbf{a} \circ f_1 \circ f_2 + \mathbf{a} \circ f_0 - \mathbf{a} \circ f_1 - \mathbf{a} \circ f_2) \cdot \mathbf{w}_1 \cdot \mathbf{w}_2. \end{aligned}$$

A nice feature of this specification is that the interpolated image $\mathbf{a}'$ is only defined over the region of interest $\mathbf{X}'$ and not over all of $\mathbb{R}^2$.

Since $f : \mathbf{Y} \rightarrow \mathbb{R}^2$, it is very likely that $f(\mathbf{Y}) \not\subset \mathbf{X}'$. This means that the image $\mathbf{b} = \mathbf{a}' \circ f$ may contain many zero values, and — if $\mathbf{Y}$ is not properly chosen — not all values of **a** will be utilized in the computation of **b**. The latter phenomenon is called *loss of information due to clipping*. A simple rotation of an image about its center provides an example of both, the introduction of zero values and loss of information due to clipping, if we choose $\mathbf{Y} = \mathbf{X}$. Figure **4.5.9** illustrates this case. Here the left image represents the input image **a** and the right image the output image $\mathbf{b} = \mathbf{a}' \circ f$. Note that the value $\mathbf{b}(y_1, y_2)$ is zero since $f(y_1, y_2) = (x_1', x_2') \notin \mathbf{X}'$. Also, the corner areas of **a** after rotation have been clipped since they do not fit into **Y**. Of course, for rotations the problem clipping is easily resolved by choosing **Y** sufficiently large.

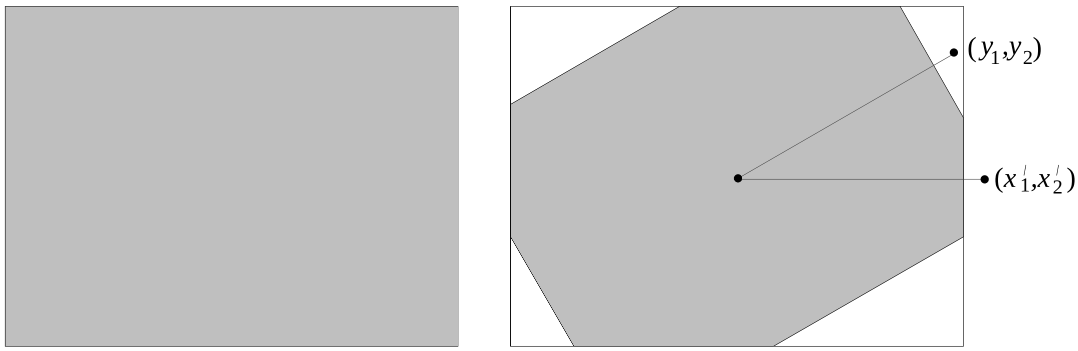

Figure 4.5.9 Rotation within the same array. The left image is the input image and the right image is the output image.

The definition of the interpolated extension $\mathbf{a}'$ requires images to be specified as computational objects rather than enumerated objects (such as the input image **a**). Once the spatial transform $f = (f_1, f_2)$ has been chosen, the dummy variables x_i and x_i' are replaced by $\lfloor f_i(y_i) \rfloor$ and $f_i(y_i)$, respectively. Each pixel value of $\mathbf{b} = \mathbf{a}' \circ f$ can then be determined pixel by pixel, line by line, or in parallel.

The spatial transformation of a digital image as exhibited by Eq. 4.5.6 represents a general scheme. It is not restricted to bilinear interpolation; any extension $\mathbf{a}'$ of $\mathbf{a}$ to a point set $\mathbf{X}'$ containing the range of *f* may be substituted. This is desirable even though bilinear interpolation and nearest neighbor approximation are the most widely used interpolation techniques. Similar to zero-order, first-order interpolation has its own drawbacks. The surface given by the graph of $\mathbf{a}'$ is not smooth; when adjacent four pixel neighborhoods are interpolated, the resulting surfaces match in amplitude at the boundaries but do not match in slope. The derivatives have, in general, discontinuities at the boundaries. In many applications, these discontinuities produce undesirable effects. In these cases, the extra additional computational cost of higher order interpolation schemes may be justified. Examples of higher order interpolation functions are cubic splines, Legendre interpolation, and $\frac{1}{x} sin x$. Higher order interpolation is usually implemented by an image-template operation.

The general formulation of spatial transformation given by Eq. 4.5.2 includes the class of affine transformations mentioned at the beginning of this section.

4.5.2 **Definition**. A transformation $f : \mathbb{R}^2 \to \mathbb{R}^2$ of the form

$$f(y_1, y_2) = (ay_1 + by_2 + v_1,\ cy_1 + dy_2 + v_2),$$

where *a*, *b*, *d*, v_1, and v_2 are constants, is called a (2–dimensional) *affine transformation*.

Observe that an equivalent definition of an affine transformation is given by

$$f(\mathbf{y}) = \mathbf{y} \cdot \mathrm{A} + \mathbf{v}\,, \tag{4.5.7}$$

where

$$\mathbf{y} = (y_1, y_2),\ \ \mathrm{A} = \begin{pmatrix} a & c \\ b & d \end{pmatrix},\ \text{ and }\ \mathbf{v} = (v_1, v_2).$$

The matrix A can always be written in the form

$$\begin{pmatrix} a & c \\ b & d \end{pmatrix} = \begin{pmatrix} r_1 \cos\theta_1 & r_1 \sin\theta_1 \\ -r_2 \sin\theta_2 & r_2 \cos\theta_2 \end{pmatrix},$$

where (r_1, θ_1) and $\left(r_2, \theta_2 + \frac{\pi}{2}\right)$ correspond to the points (a,c) and (b,d) expressed in polar form. In particular, if $r_1 = r_2$, $\theta_1 = \theta_2$, and $\mathbf{v} = \mathbf{0}$, then Eq. 4.5.7 corresponds to a rotation about the origin. If A is the identity matrix and $\mathbf{v} \neq \mathbf{0}$, then Eq. 4.5.7 represents a translation.

Various combinations of affine transformations can be used to produce highly complex patterns from simple patterns.

4.5.3 Examples:

(i) Let $f_1(\mathbf{y}) = \mathbf{y} \cdot \mathrm{A}$, where $\mathrm{A} = \begin{pmatrix} \frac{1}{\alpha} & 0 \\ 0 & \frac{1}{\alpha} \end{pmatrix}$.

If $\alpha > 1$, then $\mathbf{a} \circ f_1$ represents a magnification of **a** by the factor α (see also Example **4.5.1**). On the other hand, if $0 < \alpha < 1$, then $\mathbf{a} \circ f_1$ represents a contraction (shrinking) of **a** by the factor α. Figure **4.5.10** illustrates a contraction using the factor $\alpha = \frac{2}{3}$. Here the input image **a**, shown on the left, contains a trapezoid of base length *l* and angle of inclination θ. The output image $\mathbf{a} \circ f_1$ is shown on the right of the figure, with pixels having zero values displayed in black.

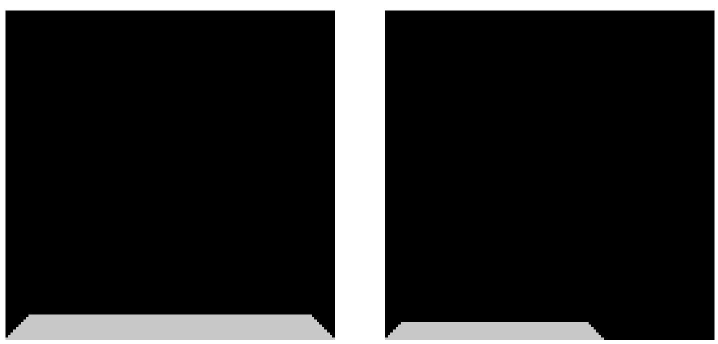

Figure 4.5.10 Image contraction using an affine map.

Suppose

$$f_2(\mathbf{y}) = \mathbf{y} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \left[(\alpha - 1)\frac{l}{2}, (\alpha - 1)\frac{l}{2} \tan\theta \right].$$

Then f_2 represents a shift in the direction θ. Composing $\mathbf{a} \circ f_1$ with f_2 results in the image $(\mathbf{a} \circ f_1) \circ f_2$ shown on the left of Figure **4.5.11**.

The composition of two affine transformations is again an affine transformation. In particular, by setting $f = f_1 \circ f_2$, it is obvious that $(\mathbf{a} \circ f_1) \circ f_2$ could have been obtained from a single affine transformation, namely

$$(\mathbf{a} \circ f_1) \circ f_2 = \mathbf{a} \circ (f_1 \circ f_2) = \mathbf{a} \circ f.$$

Now iterating the process by using the algorithm

$$\mathbf{b} := \mathbf{b} + \mathbf{b} \circ f$$

with initial $\mathbf{b} = \mathbf{a}$, results in the *railroad to infinity* shown on the right of Figure **4.5.11**.

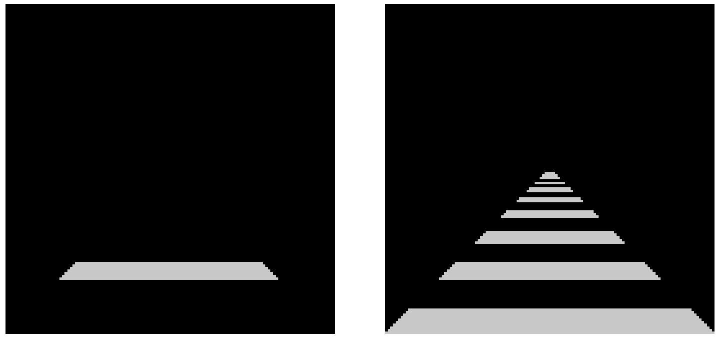

Figure 4.5.11 A railroad to infinity.

(ii) Another example of iterating affine transformations in order to create geometric patterns from simple building blocks is the construction of a brick wall from a single brick. In this example, let *w* and *l* denote the width and length of the brick shown on the left of Figure **4.5.12**. Suppose further that we want the cement layer between an adjacent pair of bricks to be of thickness *t*. A simple way of building the wall is to use two affine transformations *f* and *g*, where *f* is a horizontal shift by an amount $l + t$, and *g* is composed of a horizontal shift in the opposite direction of *f* by the amount $(l + t)/2$ and a vertical shift by the amount $w + t$. Specifically, if

$$f(y_1, y_2) = (y_1 - (l + t),\, y_2)$$

and

$$g(y_1, y_2) = (y_1 + (l + t)/2,\, y_2 - (w + t)),$$

then iterating the algorithm

$$\mathbf{a} := \mathbf{a} \vee (\mathbf{a} \circ f) \vee (\mathbf{a} \circ g)$$

will generate a brick wall whose size will depend on the number of iterations. The image on the right of Figure **4.5.12** was obtained by using the iteration

$$\begin{aligned}
&\text{repeat} \\
&\qquad \mathbf{c} := \mathbf{a} \\
&\qquad \mathbf{a} := \mathbf{a} \vee (\mathbf{a} \circ f) \vee (\mathbf{a} \circ g) \\
&\text{until } \ \mathbf{c} = \mathbf{a}
\end{aligned}$$

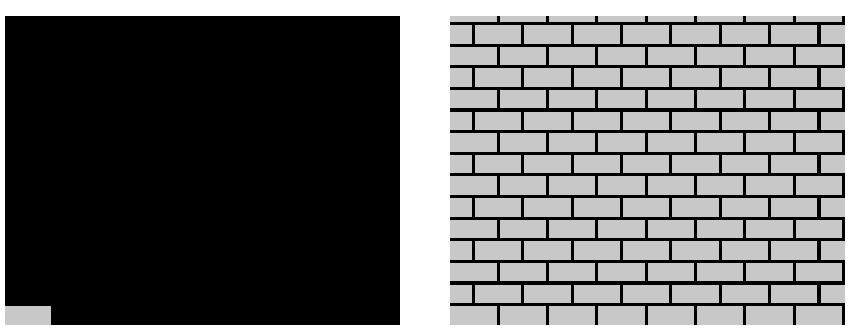

Figure 4.5.12 Generation of a brick wall from a single brick.

4.6 Set-theoretic Operations

The two prime concepts associated with a function are its domain and range. Since these two notions are viewed as mappings from $\mathbb{F}^{\mathbf{Y}}$ to $2^{\mathbf{Y}}$ and $\mathbb{F}^{\mathbf{Y}}$ to $2^{\mathbb{F}}$, they provide a link between images and their underlying point and value sets. The outputs of the domain and range functions, however, are sets and not individual points or values. In various computer vision tasks it is often necessary to extract single points or values from an image, or to know the number of elements in a particular value or point set. These tasks can be accomplished by using two fundamental concepts from elementary set theory. The two concepts are the *choice function* and the *cardinality* of a set.

The choice function, whose existence is guaranteed by axiomatic set theory [3], is not obtainable from previously defined operations. We shall view the choice function as a mapping of $2^{\mathbf{Y}} \to \mathbf{Y}$ (or $2^{\mathbb{F}} \to \mathbb{F}$) which, when applied to a set $\mathbf{Z} \subset \mathbf{Y}$, returns a randomly chosen element of A. Thus, if $\mathbf{a} \in \mathbb{N}^{\mathbf{Y}}$ and $k \in range(\mathbf{a})$, then the statement

$$choice[domain(\mathbf{a}||_k)]$$

denotes a randomly chosen pixel location whose pixel value is *k*. This example illustrates how we can get from an image back to a point in its domain. More generally, we have the following commutative diagram:

$$\begin{tikzcd} \mathbb{F}^{\mathbf{Y}} \arrow[d, "domain"'] \arrow[dr, dotted] & \\ 2^{\mathbf{Y}} \arrow[r, "choice"'] & \mathbf{Y} \end{tikzcd}$$

The dotted arrow in the diagram denotes the composition $choice \circ domain$. The choice function is, of course, not the only function from $2^{\mathbb{F}}$ to $\mathbb{F}$. For example, if $(\mathbb{F}, \gamma)$ is a commutative semigroup and $P(\mathbb{F}) = \{X \in 2^{\mathbb{F}} : \; card(X) < \infty\}$, then again we have a global reduce operation $\Gamma : P(\mathbb{F}) \to \mathbb{F}$ induced by γ and defined by $\Gamma(X) = x_1 \gamma x_2 \gamma \cdots \gamma x_n$, where $X = \{x_1, \ldots, x_n\}$. In particular, if *X* is a finite subset of $\mathbb{R}$, then $\bigvee(X)$ corresponds to the largest number in *X*.

The cardinality of a set was discussed in Section **2.7**. For finite sets the cardinality of a set corresponds to the number of elements in that set. Thus, when our discussion is restricted to digital images, we shall view the concept of cardinality as a function

$$card : 2^{\mathbb{F}} \cup 2^{\mathbf{Y}} \to \mathbb{N} \, .$$

For example, if *k* is an integer, $\mathbf{Y}$ a finite subset of $\mathbb{Z}^n$, and $\mathbf{a} \in \mathbb{N}^{\mathbf{Y}}$, then the number *n* in the statement

$$n := card[domain(\mathbf{a}||_k)]$$

corresponds to the number of pixels having value *k*. Obviously, if no pixel of **a** has value *k*, then $domain(\mathbf{a}||_k) = \varnothing$ and $n = 0$.

Note that an alternative algorithm for obtaining *n* is given by the statement

$$n := \sum \chi_{k}(\mathbf{a}),$$

and does not involve operations defined in this section.

The cardinality of a set and the choice function are unary operations *from* $2^{\mathbf{Y}}$ *to* $\mathbb{N}$ and *to* **Y**, respectively (or *from* $2^{\mathbb{F}}$ *to* $\mathbb{N}$ and $\mathbb{F}$, respectively). They do not provide for operations *on* $2^{\mathbf{Y}}$ or $2^{\mathbb{F}}$. However, there are numerous computer vision algorithms and tasks that require manipulating whole sets of pixel locations as well as sets of pixel values. In order to provide a coherent mathematical theory in which to express these types of operations more readily, we need to incorporate the Boolean algebras $\left(2^{\mathbf{Y}}, \cup, \cap, '\right)$ and $\left(2^{\mathbb{F}}, \cup, \cap, '\right)$ as part of image algebra.

The laws governing the operations on sets were given in Section **2.2** (Fig. **2.2.1**). The operations of union, intersection, and complementation can be combined to define various other set-theoretic operations. For instance, set subtraction can be defined as

$$\mathrm{A} \backslash \mathrm{B} \equiv \mathrm{A} \cap \mathrm{B}',$$

and the symmetric difference between two sets as

$$\mathrm{A} [\backslash] \mathrm{B} \equiv (\mathrm{A} \cup \mathrm{B}) \backslash (\mathrm{A} \cap \mathrm{B}).$$

It follows that $\mathrm{A} [\backslash] \mathrm{B} = (\mathrm{A} \cup \mathrm{B}) \cap (\mathrm{A} \cap \mathrm{B})' = (\mathrm{A} \cup \mathrm{B}) \cap (\mathrm{A}' \cup \mathrm{B}')$.

4.6.1 Example:

Let A and B be the two point sets in the cellular space C^2 shown in Fig. **4.6.1** (a) and (b), respectively. Thus, the set A consists of the cells whose union forms the letter “A”, while the set B consists of cells forming the letter “B”. The result of the set operations of union, intersection, set subtraction, and symmetric difference on these two sets are shown in Fig. **4.6.1** (c), (d), (e), and (f), respectively.

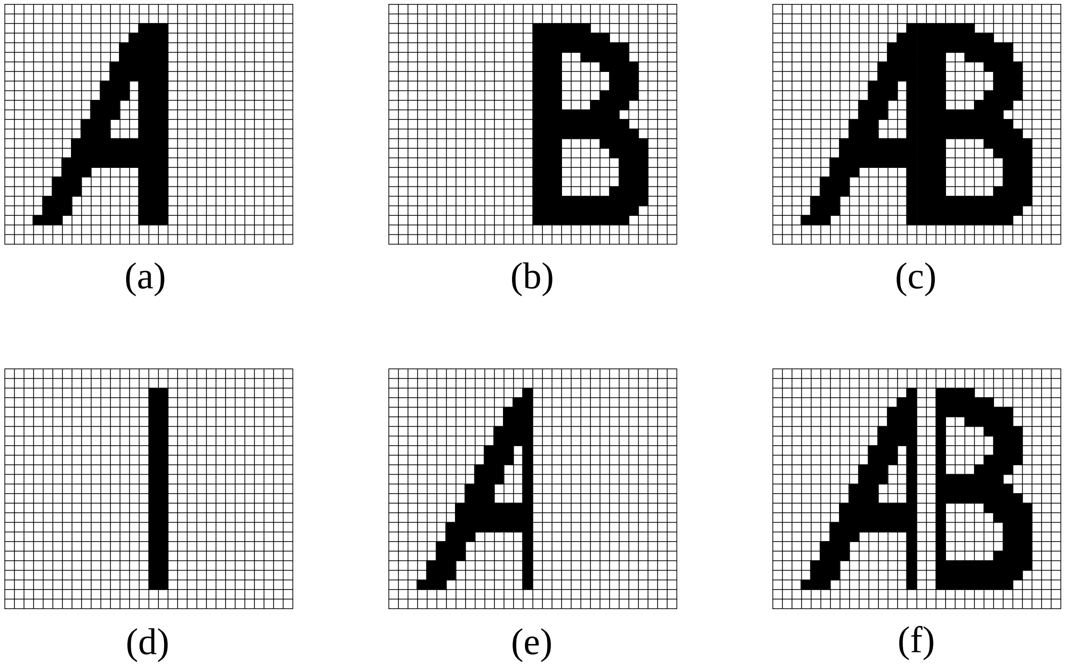

Figure 4.6.1 (a) The set A. (b) The set B. (c) The set $\mathrm{A} \cup \mathrm{B}$. (d) The set $\mathrm{A} \cap \mathrm{B}$. (e) The set $\mathrm{A} \backslash \mathrm{B}$. (f) The set $\mathrm{A}[\backslash]\mathrm{B}$.

Since $\left(2^{\mathbf{Y}}, \cup, \cap, '\right)$ and $\left(2^{\mathbb{F}}, \cup, \cap, '\right)$ are homogeneous algebras, one obvious implication is that the algebra discussed in Section **4.3** provides for the manipulation of set-valued images $\mathbf{a} : \mathbf{X} \to 2^{\mathbf{Y}}$ (or $\mathbf{a} : \mathbf{X} \to 2^{\mathbb{F}}$); the operations on set-valued images are those induced by the Boolean algebras. For example, if $\mathbf{a}, \mathbf{b} \in \left(2^{\mathbf{Y}}\right)^{\mathbf{X}}$, then

$$\mathbf{a} \cup \mathbf{b} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \cup \mathbf{b}(\mathbf{x}), \ \mathbf{x} \in \mathbf{X}\},$$

$$\mathbf{a} \cap \mathbf{b} = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \cap \mathbf{b}(\mathbf{x}), \ \mathbf{x} \in \mathbf{X}\},$$

and

$$\mathbf{a}' = \left\{(\mathbf{x}, \mathbf{c}(\mathbf{x})) : \mathbf{c}(\mathbf{x}) = [\mathbf{a}(\mathbf{x})]', \ \mathbf{x} \in \mathbf{X}\right\},$$

where $[\mathbf{a}(\mathbf{x})]' = \mathbf{Y} \backslash \mathbf{a}(\mathbf{x})$.

The Boolean image algebra $\left(\left(2^{\mathbf{Y}}\right)^{\mathbf{X}}, \cup, \cap, '\right)$ has a natural dual structure, namely $\left(\left(2^{\mathbf{X}}\right)^{\mathbf{Y}}, \cup, \cap, '\right)$. This can be ascertained from the observation that any function $\mathbf{a} : \mathbf{X} \to 2^{\mathbf{Y}}$ has a *dual* (often referred to as an *inverse* or *transpose*) $\mathbf{a}^{-1} : \mathbf{Y} \to 2^{\mathbf{X}}$, defined by

$$\mathbf{a}^{-1}(\mathbf{y}) = \{\mathbf{x} : \mathbf{x} \in \mathbf{X} \text{ and } \mathbf{y} \in \mathbf{a}(\mathbf{x})\}.$$

The function $\mathbf{a}^{-1}$ is a dual (or an inverse or transpose) in the sense that $\left(\mathbf{a}^{-1}\right)^{-1} = \mathbf{a}$.

4.6.2 **Theorem.** *The function* $\varphi : \left(2^{\mathbf{Y}}\right)^{\mathbf{X}} \to \left(2^{\mathbf{X}}\right)^{\mathbf{Y}}$ *defined by* $\varphi(\mathbf{a}) = \mathbf{a}^{-1}$ $\forall \mathbf{a} \in \left(2^{\mathbf{Y}}\right)^{\mathbf{X}}$ *is an isomorphism.*

Proof: Suppose that for some pair of functions **a** and **b** with $\mathbf{a} \neq \mathbf{b}$, we have that $\varphi(\mathbf{a}) = \varphi(\mathbf{b})$. Then $\mathbf{a} \neq \mathbf{b} \Rightarrow \exists \mathbf{x}_0 \in \mathbf{X}$ such that $\mathbf{a}(\mathbf{x}_0) \neq \mathbf{b}(\mathbf{x}_0)$. Thus, not both $\mathbf{a}(\mathbf{x}_0)$ and $\mathbf{b}(\mathbf{x}_0)$ can be empty. Suppose, without loss of generality, that $\mathbf{a}(\mathbf{x}_0) \neq \varnothing$. Since $\mathbf{a}(\mathbf{x}_0) \neq \mathbf{b}(\mathbf{x}_0)$, $\exists \mathbf{y} \in \mathbf{Y}$ such that $\mathbf{y} \in \mathbf{a}(\mathbf{x}_0)$ and $\mathbf{y} \notin \mathbf{b}(\mathbf{x}_0)$. But then $\mathbf{x}_0 \in \mathbf{a}^{-1}(\mathbf{y})$ and $\mathbf{x}_0 \notin \mathbf{b}^{-1}(\mathbf{y})$, contrary to our assumption that $\mathbf{a}^{-1} = \mathbf{b}^{-1}$. This shows that φ is one-to-one.

φ is obviously onto. Thus all that remains to be shown is that φ is a homomorphism. To show that $\varphi(\mathbf{a} \cup \mathbf{b}) = \varphi(\mathbf{a}) \cup \varphi(\mathbf{b})$ is equivalent to showing that $(\mathbf{a} \cup \mathbf{b})^{-1} = \mathbf{a}^{-1} \cup \mathbf{b}^{-1}$. Now

$$\begin{aligned}
(\mathbf{a} \cup \mathbf{b})^{-1}(\mathbf{y}) &= \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in (\mathbf{a} \cup \mathbf{b})(\mathbf{x})\} \\
&= \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in \mathbf{a}(\mathbf{x}) \cup \mathbf{b}(\mathbf{x})\} \\
&= \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in \mathbf{a}(\mathbf{x})\} \cup \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in \mathbf{b}(\mathbf{x})\} \\
&= \mathbf{a}^{-1}(\mathbf{y}) \cup \mathbf{b}^{-1}(\mathbf{y})\,.
\end{aligned}$$

In a likewise fashion we can show that $(\mathbf{a} \cap \mathbf{b})^{-1} = \mathbf{a}^{-1} \cap \mathbf{b}^{-1}$. Finally we have

$$\begin{aligned}
\left[\mathbf{a}'\right]^{-1}(\mathbf{y}) &= \left\{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in \mathbf{a}'(\mathbf{x})\right\} \\
&= \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \notin \mathbf{a}(\mathbf{x})\} \\
&= \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{y} \in \mathbf{a}(\mathbf{x})\}' \\
&= \left[\mathbf{a}^{-1}(\mathbf{y})\right]',
\end{aligned}$$

which shows that $\varphi(\mathbf{a}') = \varphi(\mathbf{a})'$.

Q.E.D.

Since typically, **Y** (or $\mathbb{F}$) is well structured — a topological space, metric space, vector space, etc. (or a semigroup, ring, field, etc.) — it makes sense to try preserve this structure by viewing **a**(**x**) as a set in **Y** (or $\mathbb{F}$) instead of a point in $2^{\mathbf{Y}}$. This change of perspective has fruitful consequences; while the theory of set-valued functions (Section **2.6**) can still be applied, intuition is maintained and properties of **Y** are not forced into hyper properties of $2^{\mathbf{Y}}$.

Borrowing from the idea of induced set-valued functions, recall that given $f : \mathbf{Y} \to \mathbf{X}$, then *f* induces a set-valued function $\check{f} : 2^{\mathbf{Y}} \to 2^{\mathbf{X}}$, which is defined as $\check{f}(\mathrm{A}) = \{f(\mathbf{x}) : \mathbf{x} \in \mathrm{A}\}$; i.e., $\check{f}(\mathrm{A}) = range(f|_{\mathrm{A}})$. In addition, there is an *inverse* $\check{f}^{-1} : 2^{\mathbf{X}} \to 2^{\mathbf{Y}}$ defined by $\check{f}^{-1}(\mathrm{B}) = \{\mathbf{y} : \mathbf{y} \in \mathbf{Y} \text{ and } f(\mathbf{y}) \in \mathrm{B}\}$. Pictorially, we have the following interesting diagram of functions:

$$\begin{tikzcd}
\mathbf{Y} \arrow[r, "f"] \arrow[d, "\mathbf{a}^{-1}"'] & \mathbf{X} \arrow[d, "\mathbf{a}"] \\
2^{\mathbf{X}} \arrow[r, "\check{f}^{-1}"] & 2^{\mathbf{Y}}
\end{tikzcd}$$

The obvious question that may arise is as to whether or not $\mathbf{a} \circ f = \check{f}^{-1} \circ \mathbf{a}^{-1}$. It can be easily verified that this is false even in the simple case where $\mathbf{X} = \mathbf{Y}$ and $f = 1_{\mathbf{X}}$. However, we do have the following theorem:

4.6.3 **Theorem.** $(\mathbf{a} \circ f)^{-1} = \check{f}^{-1} \circ \mathbf{a}^{-1}$

Proof:

$$\begin{aligned}(\mathbf{a} \circ f)^{-1}(\mathbf{y}) &= \{\mathbf{y}' \in \mathbf{Y} : \mathbf{y} \in (\mathbf{a} \circ f)(\mathbf{y}')\} \\ &= \{\mathbf{y}' \in \mathbf{Y} : \mathbf{y} \in \mathbf{a}(f(\mathbf{y}'))\} \\ &= \{\mathbf{y}' \in \mathbf{Y} : f(\mathbf{y}') \in \{\mathbf{x} \in \mathbf{X} : \mathbf{y} \in \mathbf{a}(\mathbf{x})\}\} \\ &= \check{f}^{-1}(\{\mathbf{x} \in \mathbf{X} : \mathbf{y} \in \mathbf{a}(\mathbf{x})\}) \\ &= \check{f}^{-1}(\mathbf{a}^{-1}(\mathbf{y})) \\ &= (\check{f}^{-1} \circ \mathbf{a}^{-1})(\mathbf{y}).\end{aligned}$$

.

Q.E.D.

Since φ is an isomorphism, it follows that the two images $\mathbf{a} \circ f$ and $\check{f}^{-1} \circ \mathbf{a}^{-1}$ are Boolean equivalents. Every set-valued image $\mathbf{a} : \mathbf{X} \to 2^{\mathbf{Y}}$ induces another set-valued function $\check{\mathbf{a}} : 2^{\mathbf{X}} \to 2^{\mathbf{Y}}$ defined by

$$\check{\mathbf{a}}(\mathbf{W}) = \bigcup_{\mathbf{x} \in \mathbf{W}} \mathbf{a}(\mathbf{x}).$$

4.6.4 **Example:** Let $\mathbf{a} : \mathbb{Z}^2 \to 2^{\mathbb{Z}^2}$ be given by

$$\mathbf{a}(i,j) = \{(2i,2j), (2i+1,2j), (2i,2j+1), (2i+1,2j+1)\}.$$

Then

$$\check{\mathbf{a}}(\mathbf{W}) = \bigcup_{(i,j) \in \mathbf{W}} \mathbf{a}(i,j), \text{ for } \mathbf{W} \subset \mathbb{Z}^2.$$

Figure **4.6.2** illustrates the difference between the set-valued functions **a** and $\check{\mathbf{a}}$. Note that in the illustration we view $\mathbb{Z}^2$ in terms of the cellular space C^2. The figure also illustrates the possible applications of set-valued functions for image analysis in differently scaled spaces.

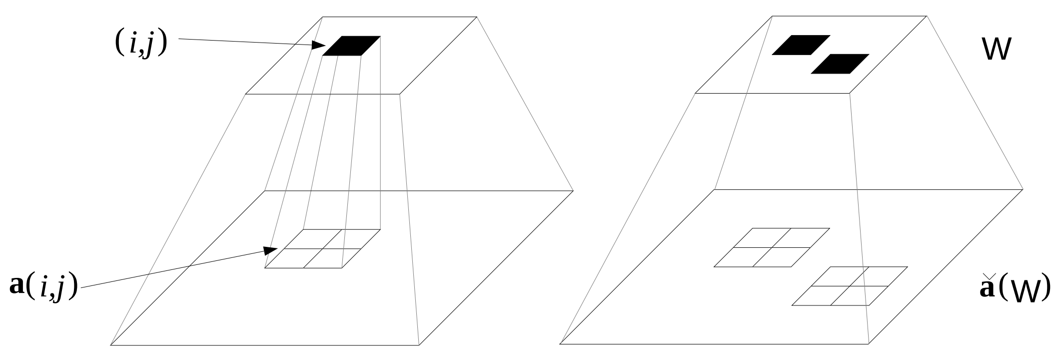

Figure 4.6.2 The set-valued function **a** and the induced function $\check{\mathbf{a}}$.

The function **a** is said to *induce a partition* on **Y** if the following two conditions are satisfied:

(i) $\mathbf{x}_1 \neq \mathbf{x}_2 \Rightarrow \mathbf{a}(\mathbf{x}_1) \cap \mathbf{a}(\mathbf{x}_2) = \emptyset$
(ii) $\mathbf{Y} = \bigcup_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x})$.

One relationship between **a**, $\mathbf{a}^{-1}$, and $\check{\mathbf{a}}$ is given by the next theorem.

4.6.5 **Theorem.** *If* $\mathbf{a}$ *induces a partition on* $\mathbf{Y}$, *then*

$$\check{\mathbf{a}} \circ \mathbf{a}^{-1}(\mathbf{Y}) = \mathbf{a}(\mathbf{X}) .$$

If only condition (i) *is satisfied, then*

$$\check{\mathbf{a}} \circ \mathbf{a}^{-1}(\mathbf{Y}) \subset \mathbf{a}(\mathbf{X}) .$$

The proof of the theorem follows directly from the definitions of the induced functions $\mathbf{a}^{-1}$ and $\check{\mathbf{a}}$, and the observation that conditions (i) and (ii) for a functionally induced partition are equivalent to conditions

(i′) $\forall \mathbf{y} \in \mathbf{Y} \; \exists$ at most one $\mathbf{x}$ s.t. $\mathbf{y} \in \mathbf{a}(\mathbf{x})$
(ii′) $\forall \mathbf{x} \in \mathbf{X} \; \exists$ at least one $\mathbf{y}$ s.t. $\mathbf{y} \in \mathbf{a}(\mathbf{x})$,

respectively.

The following example shows that, in general, $\check{\mathbf{a}} \circ \mathbf{a}^{-1}(\mathbf{Y}) \not\subset \mathbf{a}(\mathbf{X})$.

4.6.6 **Example:** Let $\mathbf{X} = \{1, 2, 3\} = \mathbf{Y}$ and $\mathbf{a} : \mathbf{X} \to 2^{\mathbf{Y}}$ be defined by

$$\mathbf{a}(i) = \begin{cases} \{1, 2\} & \text{if } i = 1 \\ \{2, 3\} & \text{if } i = 2 \\ \varnothing & \text{if } i = 3 \end{cases} .$$

Then

$$\mathbf{a}^{-1}(i) = \begin{cases} \{1\} & \text{if } i = 1 \\ \{1, 2\} & \text{if } i = 2 \\ \{2\} & \text{if } i = 3 \end{cases} .$$

Hence, $\mathbf{a}(\mathbf{X}) = \{\{1, 2\}, \{2, 3\}, \varnothing\}$. But $\mathbf{a}^{-1}(\mathbf{Y}) = \{\{1\}, \{1, 2\}, \{2\}\}$ so that

$$\begin{aligned} \check{\mathbf{a}}\big(\mathbf{a}^{-1}(\mathbf{Y})\big) &= \bigcup_{i \in \mathbf{Y}} \check{\mathbf{a}}\big(\mathbf{a}^{-1}(i)\big) \\ &= \check{\mathbf{a}}(\{1\}) \cup \check{\mathbf{a}}(\{1,2\}) \cup \check{\mathbf{a}}(\{2\}) \\ &= \{\{1\}\} \cup \{\{1, 2, 3\}\} \cup \{\{2, 3\}\} \\ &= \{\{1\}, \{1, 2, 3\}, \{2, 3\}\} . \end{aligned}$$

4.7 Operations Between Different Valued Images

The algebra $\mathbb{F}^{\mathbf{X}}$ induced by $\mathbb{F}$ is a *homogeneous image algebra*; the algebraic operations act on images of the *same* value type. To provide the capability of combining images of possibly different value types, we need to extend the operations defined on $\mathbb{F}^{\mathbf{X}}$ to a more general class of operations. This class of operations is based on the theory of generalized matrix products. Observe that Eq. 4.3.1 is practically identical to Eq. 3.14.1. The induced operation γ is a component or pixelwise operation, the only difference is that in Eq. 3.14.1 we are dealing with finite dimensional matrices. In a similar fashion we may borrow Eq. 3.14.3, the componentwise product of two matrices of different types, in order to define the pixel level operation between images of different value types.

Suppose $\mathbb{E}$, $\mathbb{F}$, and $\mathbb{G}$ are three, not necessarily distinct, value sets and $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$ a binary operation. Then $\bigcirc$ induces a binary operation

$$\bigcirc : \mathbb{E}^{\mathbf{X}} \times \mathbb{G}^{\mathbf{X}} \to \mathbb{F}^{\mathbf{X}}$$

defined as follows: for each $\mathbf{a} \in \mathbb{E}^{\mathbf{X}}$ and each $\mathbf{b} \in \mathbb{G}^{\mathbf{X}}$, define

$$\mathbf{c} = \mathbf{a} \bigcirc \mathbf{b} \in \mathbb{F}^{\mathbf{X}} \tag{4.7.1}$$

by

$$\mathbf{c}(\mathbf{x}) = \mathbf{a}(\mathbf{x}) \bigcirc \mathbf{b}(\mathbf{x}) . \tag{4.7.2}$$

Generalized unary operations are defined in a likewise fashion; any function $f : \mathbb{E} \to \mathbb{F}$ induces a unary operation $f : \mathbb{E}^{\mathbf{X}} \to \mathbb{F}^{\mathbf{X}}$ defined by

$$f(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) \,:\, \mathbf{c}(\mathbf{x}) = f(\mathbf{a}(\mathbf{x})),\ \mathbf{x} \in \mathbf{X}\}. \tag{4.7.3}$$

As an example, suppose $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$ and $f : \mathbb{R} \to \mathbb{R}^2$ is defined by $f(r) = (r, |r|)$. Then

$$f(\mathbf{a}) = \{(\mathbf{x}, \mathbf{c}(\mathbf{x})) \,:\, \mathbf{c}(\mathbf{x}) = (\mathbf{a}(\mathbf{x}), |\mathbf{a}(\mathbf{x})|),\ \mathbf{x} \in \mathbf{X}\} .$$

Whenever $(\mathbb{F}, \gamma)$ is a commutative semigroup and $\mathbf{X}$ is finite, say $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, we define the *heterogeneous image dot product* of two images as

$$\mathbf{a} \bullet \mathbf{b} = \Gamma\,(\mathbf{a} \bigcirc \mathbf{b}) ,$$

where

$$\Gamma(\mathbf{a} \bigcirc \mathbf{b}) = \mathop{\Gamma}_{\mathbf{x} \in \mathbf{X}} \mathbf{a}(\mathbf{x}) \bigcirc \mathbf{b}(\mathbf{x}) = (\mathbf{a}(\mathbf{x}_1) \bigcirc \mathbf{b}(\mathbf{x}_1))\gamma(\mathbf{a}(\mathbf{x}_2) \bigcirc \mathbf{b}(\mathbf{x}_2))\gamma \cdots \gamma(\mathbf{a}(\mathbf{x}_n) \bigcirc \mathbf{b}(\mathbf{x}_n)) . \tag{4.7.4}$$

Since the dot product involves the global reduce operation Γ, it is a binary operation whose resultant is a *scalar*; i.e., $\bullet : \mathbb{E}^{\mathbf{X}} \times \mathbb{G}^{\mathbf{X}} \to \mathbb{F}$.

Although these heterogeneous operations seem somewhat artificial, they provide a versatile environment for the specification of a wide variety of image processing tasks.

4.7.1 **Example:** (*Image centroid*) Suppose $\mathbf{X} = \{(x_1, x_2) \in \mathbb{Z}^2 \,:\, 1 \leq x_1 \leq m,\ 1 \leq x_2 \leq n\}$, $\mathbb{E} = \mathbb{R}$, $\mathbb{G} = \mathbb{Z}^2$, and $(\mathbb{F}, \gamma) = (\mathbb{R}^2, +)$. If $\bigcirc : \mathbb{R} \times \mathbb{Z}^2 \to \mathbb{R}^2$ denotes scalar multiplication of points of $\mathbb{Z}^2$ by real numbers, then $\mathbf{a} \bullet \mathbf{b} \in \mathbb{R}^2$, where $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$ and $\mathbf{b} \in (\mathbb{Z}^2)^{\mathbf{X}}$.

As a particular application example, we compute the centroid of an image. Recall that the coordinates of the center of gravity $(\bar{x}, \bar{y})$ of a continuous real-valued function *f* on a compact set $A \subset \mathbb{R}^2$ are given by

$$\bar{x} = \frac{\int\int_A x f(x,y)\,dy\,dx}{\int\int_A f(x,y)\,dy\,dx} \ \text{ and } \ \bar{y} = \frac{\int\int_A y f(x,y)\,dy\,dx}{\int\int_A f(x,y)\,dy\,dx} .$$

For $\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, the discrete approximation of these integrals is given by

$$\bar{x}_1 = \frac{\sum\limits_{x_1=1}^{m} \sum\limits_{x_2=1}^{n} x_1 \mathbf{a}(x_1, x_2)}{\sum\limits_{x_1=1}^{m} \sum\limits_{x_2=1}^{n} \mathbf{a}(x_1, x_2)} \ \text{ and } \ \bar{x}_2 = \frac{\sum\limits_{x_1=1}^{m} \sum\limits_{x_2=1}^{n} x_2 \mathbf{a}(x_1, x_2)}{\sum\limits_{x_1=1}^{m} \sum\limits_{x_2=1}^{n} \mathbf{a}(x_1, x_2)} .$$

Considering these formulas, it is not difficult to ascertain that the image centroid $\bar{\mathbf{x}} = (\bar{x}_1, \bar{x}_2)$ is given by the simple image algebra statement

$$\bar{\mathbf{x}} := \frac{\mathbf{a} \bullet \mathbf{i}}{\sum \mathbf{a}},$$

where $\mathbf{i} : \mathbf{X} \to \mathbb{Z}^2$ denotes the inclusion map $\mathbf{i}(x_1, x_2) = (x_1, x_2)$. Figure **4.7.1** provides an illustration of an image centroid. The Boolean source image **a**, which shows the sillhouette of an SR71 spy plane, is shown on the left. The image on the right shows the location of the centroid which is in the center of the von Neumann configuration.

Figure 4.7.1 Example of an image centroid; the source image **a** is on the left, the location of the centroid is the center of the von Neumann configuration in the image on the right.

4.8 Image-Template Products

Image-template products are one of the most useful consequences of the concept of a heterogeneous image product, they provide the rules for combining images with templates and templates with templates. These products include the usual correlation and convolution products used in digital signal processing.

If $\mathbf{t} \in \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}}$, then for each $\mathbf{y} \in \mathbf{Y}$, $\mathbf{t}_{\mathbf{y}} \in \mathbb{G}^{\mathbf{X}}$. Thus, if $\mathbf{a} \in \mathbb{E}^{\mathbf{X}}$, then according to Eqs. 4.7.1, 4.7.2, and 4.7.4, $\mathbf{a} \bigcirc \mathbf{t}_{\mathbf{y}} \in \mathbb{F}^{\mathbf{X}}$ and $\Gamma(\mathbf{a} \bigcirc \mathbf{t}_{\mathbf{y}}) \in \mathbb{F}$. It follows that the binary operations $\bigcirc$ and γ induce a binary operation

$$\textcircled{\gamma} : \mathbb{E}^{\mathbf{X}} \times \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}} \to \mathbb{F}^{\mathbf{Y}},$$

where

$$\mathbf{b} = \mathbf{a} \textcircled{\gamma} \mathbf{t} \in \mathbb{F}^{\mathbf{Y}}$$

is defined by

$$\begin{aligned} \mathbf{b}(\mathbf{y}) = \Gamma(\mathbf{a} \bigcirc \mathbf{t}_{\mathbf{y}}) &= \mathop{\Gamma}_{\mathbf{x} \in \mathbf{X}} (\mathbf{a}(\mathbf{x}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x})) \\ &= (\mathbf{a}(\mathbf{x}_1) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_1))\gamma(\mathbf{a}(\mathbf{x}_2) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_2))\gamma \cdots \gamma(\mathbf{a}(\mathbf{x}_n) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_n)). \end{aligned} \tag{4.8.1}$$

The expression $\mathbf{a} \textcircled{\gamma} \mathbf{t}$ is called the *right product of* **a** *with* **t**. Note that Eq. 4.8.1 is essentially Eq. 3.14.16 with **x** and **y** replacing *k* and *t*, respectively.

Every template $\mathbf{s} \in \left(\mathbb{G}^{\mathbf{Y}}\right)^{\mathbf{X}}$ has a transpose $\mathbf{s}' \in \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}}$ which is defined $\mathbf{s}'_{\mathbf{y}}(\mathbf{x}) = \mathbf{s}_{\mathbf{x}}(\mathbf{y})$. Obviously, $(\mathbf{s}')' = \mathbf{s}$ and $\mathbf{s}'$ reverses the mapping order from $\mathbf{X} \to \mathbb{G}^{\mathbf{Y}}$ to $\mathbf{Y} \to \mathbb{G}^{\mathbf{X}}$. If we also reverse the order of $\mathbb{E}$ and $\mathbb{G}$ in our previous discussion by assuming $\bigcirc : \mathbb{G} \times \mathbb{E} \to \mathbb{F}$, then we have $\mathbf{s}'_{\mathbf{y}} \bigcirc \mathbf{a} \in \mathbb{F}^{\mathbf{X}}$ and $\Gamma\left(\mathbf{s}'_{\mathbf{y}} \bigcirc \mathbf{a}\right) \in \mathbb{F}$, for $\mathbf{a} \in \mathbb{E}^{\mathbf{X}}$ and $\mathbf{s} \in \left(\mathbb{G}^{\mathbf{Y}}\right)^{\mathbf{X}}$. Hence the binary operations $\bigcirc$ and γ induce a binary operation

$$\textcircled{\gamma} : \left(\mathbb{G}^{\mathbf{Y}}\right)^{\mathbf{X}} \times \mathbb{E}^{\mathbf{X}} \to \mathbb{F}^{\mathbf{Y}} ,$$

where

$$\mathbf{b} = \mathbf{s} \textcircled{\gamma} \mathbf{a} \in \mathbb{F}^{\mathbf{Y}}$$

is defined by

$$\mathbf{b}(\mathbf{y}) = \Gamma\left(\mathbf{s}'_{\mathbf{y}} \bigcirc \mathbf{a}\right) = \Gamma_{\mathbf{x} \in \mathbf{X}} \left(\mathbf{s}'_{\mathbf{y}}(\mathbf{x}) \bigcirc \mathbf{a}(\mathbf{x})\right) . \tag{4.8.2}$$

The expression $\mathbf{s} \textcircled{\gamma} \mathbf{a}$ is called the *left product of* **a** *with* **s**. Here Eq. 4.8.2 is the analogue of Eq. 3.14.17. Observe that for both the left and the right product, the source image **a** is an $\mathbb{E}$–valued image on **X**, while the transformed image **b** is an $\mathbb{F}$–valued image on **Y**. It follows that image–template products are capable of performing both pixel level and spatial level image transformations; images with certain range values defined over a given point set are transformed into images with entirely different range values over point sets of possibly different spaces.

When computing $\mathbf{s} \textcircled{\gamma} \mathbf{a}$, it is not necessary to use the transpose $\mathbf{s}'$ since

$$\Gamma_{\mathbf{x} \in \mathbf{X}} \left(\mathbf{s}'_{\mathbf{y}}(\mathbf{x}) \bigcirc \mathbf{a}(\mathbf{x})\right) = \Gamma_{\mathbf{x} \in \mathbf{X}} \left(\mathbf{s}_{\mathbf{x}}(\mathbf{y}) \bigcirc \mathbf{a}(\mathbf{x})\right) . \tag{4.8.3}$$

This allows us to redefine the transformation $\mathbf{b} = \mathbf{s} \textcircled{\gamma} \mathbf{a}$ as

$$\mathbf{b}(\mathbf{y}) = \Gamma_{\mathbf{x} \in \mathbf{X}} \left(\mathbf{s}_{\mathbf{x}}(\mathbf{y}) \bigcirc \mathbf{a}(\mathbf{x})\right) . \tag{4.8.4}$$

Image–template products are computation intensive. For example, if **X** is an $m \times m$ array of points and **Y** of size $n \times n$, then according to Eq. 4.8.1, the computation of each image value **b**(**y**) requires m^2 operations of type $\bigcirc$ and $m^2 - 1$ operations of type γ, for a total of $2m^2 - 1$ operations. Thus, in order to construct the image **b**, a total number of $n^2\left(2m^2 - 1\right)$ are required. For standard 512×512 or 1024×1024 size images computations of such magnitude would be prohibitive even on todays supercomputers. Fortunately, in most cases $(\mathbb{F}, \gamma)$ is a monoid. Hence, whenever $\mathbf{a}(\mathbf{x}_i) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_i) = 0$, where 0 denotes the zero of $(\mathbb{F}, \gamma)$, the two γ operations

$$(\mathbf{a}(\mathbf{x}_{i-1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i-1}))\gamma(\mathbf{a}(\mathbf{x}_i) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_i))\gamma(\mathbf{a}(\mathbf{x}_{i+1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i+1}))$$

collapse into a single γ operation since

$$\begin{aligned}&(\mathbf{a}(\mathbf{x}_{i-1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i-1}))\gamma(\mathbf{a}(\mathbf{x}_i) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_i))\gamma(\mathbf{a}(\mathbf{x}_{i+1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i+1})) \\ &= (\mathbf{a}(\mathbf{x}_{i-1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i-1}))\gamma(\mathbf{a}(\mathbf{x}_{i+1}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}_{i+1})) .\end{aligned}$$

For the remainder of this section we assume that $(\mathbb{F}, \gamma)$ is a monoid and let 0 denote the zero of $\mathbb{F}$ under the operation γ. Suppose $\mathbf{a} \in \mathbb{E}^{\mathbf{X}}$ and $\mathbf{t} \in \left(\mathbb{G}^{\mathbf{Z}}\right)^{\mathbf{Y}}$, where **X** and **Z** are subsets of the same topological space. Since $\mathbb{F}$ is a monoid, the operator $\textcircled{\gamma}$ can be extended to a mapping

$$\textcircled{\gamma} : \mathbb{E}^{\mathbf{X}} \times \left(\mathbb{G}^{\mathbf{Z}}\right)^{\mathbf{Y}} \to \mathbb{F}^{\mathbf{Y}} ,$$

where $\mathbf{b} = \mathbf{a} \,\textcircled{\gamma}\, \mathbf{t}$ is defined by

$$\mathbf{b}(\mathbf{y}) = \begin{cases} \underset{\mathbf{x} \in \mathbf{X} \cap \mathbf{Z}}{\Gamma} \left(\mathbf{a}(\mathbf{x}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x})\right) & \text{if } \mathbf{X} \cap \mathbf{Z} \neq \varnothing \\ 0 & \text{if } \mathbf{X} \cap \mathbf{Z} = \varnothing \, . \end{cases} \tag{4.8.5}$$

The left product $\mathbf{s} \,\textcircled{\gamma}\, \mathbf{a}$ is defined in a similar fashion. Subsequent examples will demonstrate that the ability of replacing **X** with **Z** greatly simplifies the use of templates in algorithm development.

Significant reduction in the number of computations involving the image–template product can be achieved if $\mathbb{E} = \mathbb{G} = \mathbb{F}$ is a monoid and $(\mathbb{F}, \gamma, \bigcirc)$ is a commutative semiring. If $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{Z}}\right)^{\mathbf{Y}}$, then the *support of* $\mathbf{t}$ *at a point* $\mathbf{y} \in \mathbf{Y}$ *with respect to the operation* γ is defined as $S_0(\mathbf{t}_{\mathbf{y}}) = \{\mathbf{x} \in \mathbf{Z} \, : \, \mathbf{t}_{\mathbf{y}}(\mathbf{x}) \neq 0\}$. Since $\mathbf{t}_{\mathbf{y}}(\mathbf{x}) = 0$ whenever $\mathbf{x} \notin S_0(\mathbf{t}_{\mathbf{y}})$, we have that $\mathbf{a}(\mathbf{x}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x}) = 0$ whenever $\mathbf{x} \notin S_0(\mathbf{t}_{\mathbf{y}})$ and, therefore,

$$\underset{\mathbf{x} \in \mathbf{X} \cap \mathbf{Z}}{\Gamma} \left(\mathbf{a}(\mathbf{x}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x})\right) = \underset{\mathbf{x} \in \mathbf{X} \cap S_0(\mathbf{t}_{\mathbf{y}})}{\Gamma} \left(\mathbf{a}(\mathbf{x}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x})\right) . \tag{4.8.6}$$

It follows that the computation of the new pixel value **b(y)** does not depend on the size of **X**, but on the size of $S_0(\mathbf{t}_{\mathbf{y}})$. Therefore, if $k = card(\mathbf{X} \cap S_0(\mathbf{t}_{\mathbf{y}}))$, then the computation of **b(y)** requires a total of $2k - 1$ operations of type γ and $\bigcirc$. If *m* and *n* are as above, and *k* is small, then $n^2(2k-1)$ is significantly smaller than $n^2\left(m^2 - 1\right)$.

Substitution of different value sets and specific binary operations for γ and $\bigcirc$ results in a wide variety of different image transforms. Our prime examples will be the ring $(\mathbb{R}, +, \cdot)$ and the bounded *l*-groups $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$ and $\left(\mathbb{R}_{\infty}^{\geq 0}, \vee, \wedge, \times, \times'\right)$. Here we substitute $\mathbb{R}_{\infty}^{\geq 0} = \mathbb{R}^+ \cup \{0, \infty\}$ for the bounded *l*-group $\left(\mathbb{R}_{\pm\infty}^{+}, \vee, \wedge, \times, \times'\right)$ discussed in Section **3.12** by replacing the symbol $-\infty$ with 0. From a mathematical perspective, this amounts only to an interchange of symbols. However, from a practical standpoint, this interchange allows for the manipulation of non-negative real-valued images that may contain zero values.

Replacing $(\mathbb{F}, \gamma, \bigcirc)$ by $(\mathbb{R}, +, \cdot)$ changes $\mathbf{b} = \mathbf{a} \,\textcircled{\gamma}\, \mathbf{t}$ into

$$\mathbf{b} = \mathbf{a} \oplus \mathbf{t} \, ,$$

where

$$\mathbf{b}(\mathbf{y}) = \sum_{\mathbf{x} \in \mathbf{X} \cap S(\mathbf{t}_{\mathbf{y}})} \left(\mathbf{a}(\mathbf{x}) \cdot \mathbf{t}_{\mathbf{y}}(\mathbf{x})\right) , \tag{4.8.7}$$

$\mathbf{a} \in \mathbb{R}^{\mathbf{X}}$, and $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{Z}}\right)^{\mathbf{Y}}$. The left product is defined in a similar fashion. In digital image processing **X** and **Y** are usually arrays of form $\{(i, j) \, : \, 1 \leq i \leq m, \; 1 \leq j \leq n\}$. In this case the relationship between the right and left product can be stated in terms of transposes:

$$\mathbf{a} \oplus \mathbf{t} = \left(\mathbf{t}' \oplus \mathbf{a}'\right)' ,$$

where $\mathbf{a}'(i, j) = \mathbf{a}(j, i)$.

4.8.1 **Example:** (*Local Averaging*) Let **a** be a real-valued image on a rectangular array $\mathbf{X} \subset \mathbb{Z}^2$ and $\mathbf{t} \in \left(\mathbb{R}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ the translation invariant 3×3 neighborhood template defined by

$$\mathbf{t}_{\mathbf{y}} = \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

The image $\mathbf{b}$ obtained from the code

$$\mathbf{b} := \frac{1}{9}(\mathbf{a} \oplus \mathbf{t})$$

represents the image obtained from $\mathbf{a}$ by local averaging since the new pixel value $\mathbf{b}(\mathbf{y})$ is given by

$$\mathbf{b}(\mathbf{y}) = \frac{1}{9} \sum_{\mathbf{x} \in \mathbf{X} \cap S(\mathbf{t_y})} \mathbf{a}(\mathbf{x}) \cdot \mathbf{t_y}(\mathbf{x}) = \frac{1}{9} \sum_{\mathbf{x} \in \mathbf{X} \cap S(\mathbf{t_y})} \mathbf{a}(\mathbf{x}) .$$

Figure **4.8.1** illustrates the effect of local averaging. The source image $\mathbf{a}$ is on the left and the locally averaged image is on the right.

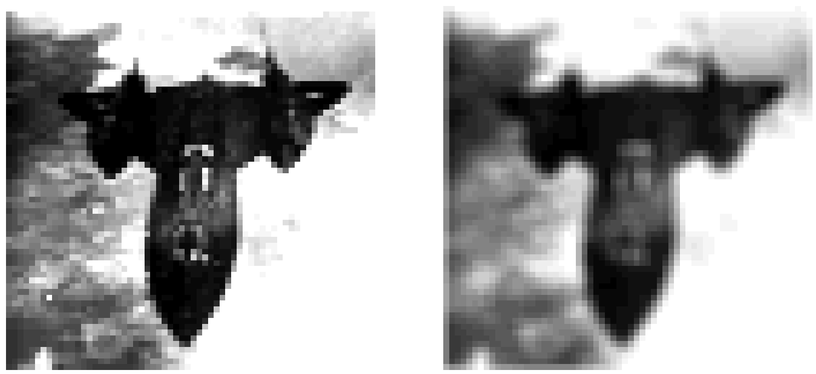

Figure 4.8.1 Effect of local averaging. The source image is displayed on the left and the averaged image on the right.

It is important to distinguish between the mathematical equality $\mathbf{b} = \mathbf{a} \oplus \mathbf{t}$ and the program code representation $\mathbf{b} := \mathbf{a} \oplus \mathbf{t}$. The former denotes equality of two images, while the latter means to replace $\mathbf{b}$ by $\mathbf{a} \oplus \mathbf{t}$. In Example **4.8.1**, the image $\frac{1}{9}(\mathbf{a} \oplus \mathbf{t})$ is, by definition, an image on all of $\mathbb{Z}^2$ with zero values outside the array $\mathbf{X} \subset \mathbb{Z}^2$. This is not a problem if we represent $\frac{1}{9}(\mathbf{a} \oplus \mathbf{t})$ as a function. However, if we represent $\frac{1}{9}(\mathbf{a} \oplus \mathbf{t})$ as a data structure, such as an array of numbers, then we run into the problem of storing an infinite array on a finite machine. Also, in practice, one is only interested in the image $\frac{1}{9}(\mathbf{a} \oplus \mathbf{t})$ restricted to the array $\mathbf{Y}$ (in our particular example $\mathbf{Y} = \mathbf{X}$) on which $\mathbf{b}$ is defined, that is $\frac{1}{9}(\mathbf{a} \oplus \mathbf{t})|_{\mathbf{Y}}$.

This problem could be solved as follows: Let $\mathbf{s} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{X}}$ be defined by $\mathbf{s_y} \equiv (\mathbf{t_y})|_{\mathbf{X}}$ for each $\mathbf{y} \in \mathbf{X}$, where $\mathbf{t}$ is the template defined in Example 4.8.1. Then $\frac{1}{9}(\mathbf{a} \oplus \mathbf{s})$ provides the desired digital image since $\mathbf{a} \oplus \mathbf{s} = (\mathbf{a} \oplus \mathbf{t})|_{\mathbf{X}}$. Thus we may ask "why not define $\mathbf{t}$ simply as a template from $\mathbf{X}$ to $\mathbf{X}$ instead from $\mathbb{Z}^2$ to $\mathbb{Z}^2$?" The answer is that by defining the local averaging template as we did, the template can be used for smoothing *any* two-dimensional image, independent on the dimension of $\mathbf{X}$. The reason for this is that when defining an image $\mathbf{b}$ in a program, one usually has to declare its dimensions, i.e., the size of its underlying array. In particular, if $\mathbf{b}$ is declared to be an image on $\mathbf{Y}$, then the image algebra pseudo code $\mathbf{b} := \mathbf{a} \oplus \mathbf{t}$ means to replace $\mathbf{b}$ pointwise by $\mathbf{a} \oplus \mathbf{t}$ such that the value of $\mathbf{b}$ at point $\mathbf{y}$ is the value of $\mathbf{a} \oplus \mathbf{t}$ at point $\mathbf{y}$, where, of course, $\mathbf{Y} \subset \mathbb{Z}^2$. In terms of algebraic equality, we then have $\mathbf{b} = (\mathbf{a} \oplus \mathbf{t})|_{\mathbf{Y}}$. As a result, a programmer is not faced with the task of redefining $\mathbf{t}$ for different-sized images as would have been the case if we had defined $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{X}}$.

Of course, the program statement $\mathbf{b} := \mathbf{a} \oplus \mathbf{t}$ will produce a boundary effect. In particular, if $\mathbf{a}$ and $\mathbf{b}$ are $m \times n$ images with underlying coordinate set $\mathbf{X} = \left\{(i,j) \in \mathbb{Z}^2 \,:\, 1 \leq i \leq m,\ 1 \leq j \leq n\right\}$, then $\mathbf{b}(1,1) = \frac{1}{9}(\mathbf{a}(1,1) + \mathbf{a}(1,2) + \mathbf{a}(2,1) + \mathbf{a}(2,2))$, which is not the average of four points. One may either ignore this boundary effect (the most common choice) or use one of several schemes to prevent it. For instance, one may simply avoid boundary pixels by defining an array $\mathbf{Z} = \{(i,j) \in \mathbb{Z}^2 : 1 < i < m,\ 1 < j < n\}$ and setting $\mathbf{b} := \left[\frac{1}{9}(\mathbf{a} \bigoplus \mathbf{t})\right]|_{\mathbf{Z}}$. Then **b** represents the desired $(m-2) \times (n-2)$ output image. Letting *m* and *n* be variables allows the application of **t** to images of arbitrary size.

If, on the other hand, a true average is desired in the interior as well as along the image boundary, then an appropriate variant template can be defined to accomplish this task. Specifically, for $\mathbf{y} = (y_1, y_2) \in \mathbb{Z}^2$, set $N(\mathbf{y}) = \{(y_1 + i, y_2 + j) : i,j \in \{-1,0,1\}\}$ and define **t** by

$$\mathbf{t}_{\mathbf{y}}(\mathbf{x}) = \begin{cases} \frac{1}{4} & \text{if } \mathbf{y} \in \{(1,1),(1,n),(m,1),(m,n)\} \text{ and } \mathbf{x} \in N(\mathbf{y}) \cap \mathbf{X} \\ \frac{1}{6} & \text{if } \mathbf{y} \in \{(i,1),(1,j),(i,n),(m,j) : 1 < i < m,\ 1 < j < n\} \\ & \text{and } \mathbf{x} \in N(\mathbf{y}) \cap \mathbf{X} \\ \frac{1}{9} & \text{if } \mathbf{y} \in \{(i,j) : 1 < i < m,\ 1 < j < n\} \text{ and } \mathbf{x} \in N(\mathbf{y}) \cap \mathbf{X} \\ 0 & \text{otherwise}. \end{cases}$$

In Section **4.5** we hinted that spatial transformations may also be performed using template operations. This should come as no great surprise since templates operate on both pixel values and pixel locations. We provide an example of a spatial transform in terms of a left image–template product.

4.8.2 Example: (*Image rotation*) Let **a** be a real-valued image on a rectangular array $\mathbf{X} = \{(x_1, x_2) \in \mathbb{Z}^2 : 1 \le x_1 \le m,\ 1 \le x_2 \le n\}$, $P = \{\theta : -2\pi < \theta < 2\pi\}$, and $\mathbf{r} : P \to \left(\mathbb{R}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ a parametrized template defined by

$$\mathbf{r}(\theta)_{(x_1,x_2)}(y_1,y_2) = \begin{cases} 1 & \text{if } (y_1,y_2) = ([x_1 cos\theta - x_2 sin\theta], [x_1 sin\theta + x_2 cos\theta]) \\ 0 & \text{otherwise} \end{cases}$$

Then the image **b** obtained by setting

$$\mathbf{b} := \mathbf{r}(\theta) \bigoplus \mathbf{a}$$

represents **a** rotated through an angle θ using nearest neighbor interpolation. In this example the symbol $[a]$ denotes the rounding of *a* to the nearest integer.

The bounded *l*-group $(\mathbb{R}_{\pm\infty}, \vee, \wedge, +, +')$ provides for two lattice products:

$$\mathbf{b} = \mathbf{a} \boxed{\vee} \mathbf{t},$$

where

$$\mathbf{b}(\mathbf{y}) = \bigvee_{\mathbf{x} \in \mathbf{X} \cap S_{-\infty}(\mathbf{t}_{\mathbf{y}})} [\mathbf{a}(\mathbf{x}) + \mathbf{t}_{\mathbf{y}}(\mathbf{x})], \tag{4.8.8}$$

and

$$\mathbf{b} = \mathbf{a} \boxed{\wedge} \mathbf{t},$$

where

$$\mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X} \cap S_{\infty}(\mathbf{t}_{\mathbf{y}})} [\mathbf{a}(\mathbf{x}) +' \mathbf{t}_{\mathbf{y}}(\mathbf{x})]. \tag{4.8.9}$$

In order to distinguish between these two types of lattice transforms, we call the operator $\boxed{\vee}$ the *additive maximum* and $\boxed{\wedge}$ the *additive minimum*. It follows from our earlier discussion that if $\mathbf{X} \cap S_{-\infty}(\mathbf{t}_{\mathbf{y}}) = \varnothing$, then the value of **b(y)** in Eq. 4.8.8 is $-\infty$, the *zero* of $\mathbb{R}_{\pm\infty}$ under the operation of $\vee$. Similarly, if $\mathbf{X} \cap S_{\infty}(\mathbf{t}_{\mathbf{y}}) = \varnothing$, then $\mathbf{b}(\mathbf{y}) = \infty$ in Eq. 4.8.9.

The left additive max and min operations are defined by

$$\mathbf{t} \boxed{\vee} \mathbf{a} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \,:\, \mathbf{b}(\mathbf{y}) = \bigvee_{\mathbf{x} \in \mathbf{X} \cap S_{-\infty}(\mathbf{t}_{\mathbf{y}})} [\mathbf{t}_{\mathbf{x}}(\mathbf{y}) + \mathbf{a}(\mathbf{x})] \,,\ \mathbf{y} \in \mathbf{Y} \right\} \tag{4.8.10}$$

and

$$\mathbf{t} \boxed{\wedge} \mathbf{a} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \,:\, \mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X} \cap S_{\infty}(\mathbf{t}_{\mathbf{y}})} \left[\mathbf{t}_{\mathbf{x}}(\mathbf{y}) +' \mathbf{a}(\mathbf{x})\right] \,,\ \mathbf{y} \in \mathbf{Y} \right\}, \tag{4.8.11}$$

respectively. The relationship between additive max and min is given in terms of lattice duality by

$$\mathbf{a} \boxed{\wedge} \mathbf{t} = \left(\mathbf{t}^* \boxed{\vee} \mathbf{a}^*\right)^*,$$

where the image $\mathbf{a}^*$ is defined by $\mathbf{a}^*(\mathbf{x}) = [\mathbf{a}(\mathbf{x})]^*$, and the conjugate (or dual) of $\mathbf{t} \in \left(\mathbb{R}_{\pm\infty}^{\mathbf{X}}\right)^{\mathbf{Y}}$ is the template $\mathbf{t}^* \in \left(\mathbb{R}_{\pm\infty}^{\mathbf{Y}}\right)^{\mathbf{X}}$ defined by $\mathbf{t}^*_{\mathbf{x}}(\mathbf{y}) = [\mathbf{t}_{\mathbf{y}}(\mathbf{x})]^*$. It follows that $\mathbf{t}^*_{\mathbf{x}}(\mathbf{y}) = -\mathbf{t}'_{\mathbf{y}}(\mathbf{x})$.

4.8.3 Example: (*Salt and pepper noise removal*) An image may be subject to noise and interference from several sources. Image noise arising from a noisy sensor or channel transmission errors usually appears as discrete isolated pixel variations that are not spatially correlated. Pixels that are in error often appear markedly different from their neighbors. In Boolean images this type of noise appears as isolated black and white pixels and is referred to as *salt and pepper noise*. The image **a** on the top left of Figure **4.8.2** provides an example of this type of noise. The silhouette of the SR 71 spy plane contains small holes while the background contains isolated black pixels or small isolated groups of black pixels. One way of removing this type of salt and pepper noise is by geometric filtering using local maxima and minima.

Let $\mathbb{F}$ denote the 3–element bounded subgroup $\{-\infty, 0, \infty\}$ of $\mathbb{R}_{\pm\infty}$, and $\mathbf{t} \in \left(\mathbb{F}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ be defined by

$$\mathbf{t}_{(y_1,y_2)}(x_1,x_2) = \begin{cases} 0 & \text{if } (x_1,x_2) \in \{(y_1 \pm 1, y_2), (y_1, y_2 \pm 1)\} \\ -\infty & \text{otherwise}. \end{cases}$$

Pictorially, $\mathbf{t}$ can be represented as

$$\mathbf{t}_{\mathbf{y}} = \begin{array}{|c|c|c|} \hline & 0 & \\ \hline 0 & & 0 \\ \hline & 0 & \\ \hline \end{array}$$

where the cells outside the support of $\mathbf{t}_{\mathbf{y}}$ (everything not labelled zero) have value $-\infty$.

If $\mathbf{b} := \mathbf{a} \boxed{\vee} \mathbf{t}$, then

$$\begin{aligned} \mathbf{b}(y_1,y_2) &= \bigvee_{(x_1,x_2) \in S_{-\infty}\left(\mathbf{t}_{(y_1,y_2)}\right)} \left[\mathbf{a}(x_1,x_2) + \mathbf{t}_{(y_1,y_2)}(x_1,x_2)\right] \\ &= \bigvee_{(x_1,x_2) \in \{(y_1 \pm 1, y_2), (y_1, y_2 \pm 1)\}} [\mathbf{a}(x_1,x_2) + 0] \\ &= \mathbf{a}(y_1+1,y_2) \vee \mathbf{a}(y_1-1,y_2) \vee \mathbf{a}(y_1,y_2+1) \vee \mathbf{a}(y_1,y_2-1)\,. \end{aligned}$$

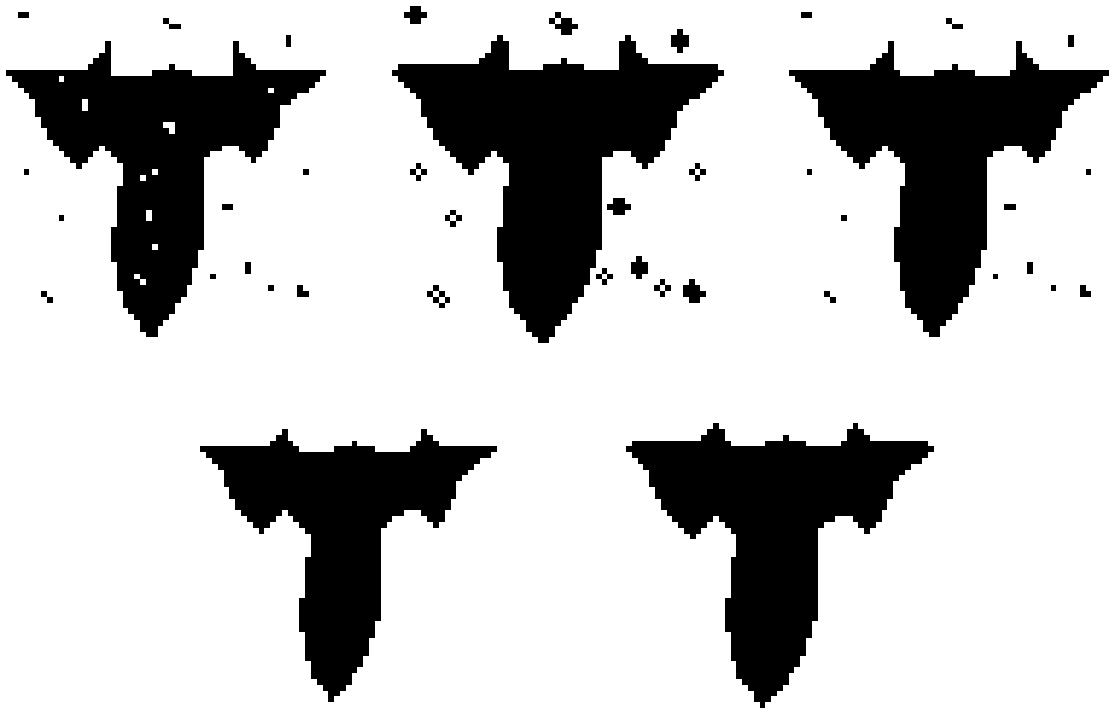

Figure 4.8.2 Example of salt and pepper noise removal. The sequence of images illustrates the different steps of the algorithm, starting with the noisy source image on the left in the top row and ending with the cleaned image shown on the bottom right.

Thus, **b**(**y**) is the maximum of **a** restricted to a von Neumann neighborhood about the point **y**. In particular, if $\mathbf{a}(\mathbf{y}) = 0$ but $\mathbf{a}(\mathbf{x}) = 1$ for any 4–neighbor **x** of **y**, then $\mathbf{b}(\mathbf{y}) = 1$. Therefore, in the transformed image **b**, any small hole appearing in **a** will have been filled in as illustrated by the top center image of Figure **4.8.2**. However, since we are taking local maxima, all black objects in **a** have become enlarged or *dilated*. To shrink black objects back to their former size, we convolve **b** with $\mathbf{t}^*$ using the additive minimum operator $\boxed{\wedge}$. The conjugate $\mathbf{t}^*$ of **t** *looks the same* as **t**, the only difference is that pixel values of $-\infty$ have been replaced with ∞. The resulting image $\mathbf{c} := \mathbf{b} \boxed{\wedge} \mathbf{t}^*$ is shown on the top right of Figure **4.8.2**.

In order to remove the pepper noise, we need to apply the additive min operator one more time. The image $\mathbf{d} := \mathbf{c} \boxed{\wedge} \mathbf{t}^*$ is shown on the bottom left of Fig. **4.8.2**. Unfortunately, the second application of the operator $\boxed{\wedge}$ also eroded the silhouette of the SR 71. To dilate the eroded silhouette back to its former size we need to apply the additive max operator to **d**. This results in the final filtered image $\mathbf{f} := \mathbf{d} \boxed{\vee} \mathbf{t}$ shown on the bottom right of Figure **4.8.2**. The complete algorithm resulting in **f** can be stated in one line of pseudo-code; namely,

$$\mathbf{f} := (((\mathbf{a} \boxed{\vee} \mathbf{t}) \boxed{\wedge} \mathbf{t}^*) \boxed{\wedge} \mathbf{t}^*) \boxed{\vee} \mathbf{t}\,.$$

The bounded *l*-group $\left(\mathbb{R}_{\infty}^{\geq 0}, \vee, \wedge, \times, \times'\right)$ also provides for two lattice products. Specifically, we have

$$\mathbf{b} = \mathbf{a} \textcircled{\vee} \mathbf{t}\,,$$

where

$$\mathbf{b}(\mathbf{y}) = \bigvee_{\mathbf{x} \in \mathbf{X} \cap S(\mathbf{t_y})} [\mathbf{a}(\mathbf{x}) \times \mathbf{t_y}(\mathbf{x})] \, , \tag{4.8.12}$$

and

$$\mathbf{b} = \mathbf{a} \textcircled{\scriptsize$\wedge$} \mathbf{t} \, ,$$

where

$$\mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X} \cap S_\infty(\mathbf{t_y})} \left[\mathbf{a}(\mathbf{x}) \times' \mathbf{t_y}(\mathbf{x})\right] . \tag{4.8.13}$$

Here 0 is the zero of $\mathbb{R}^{\geq 0}_{\infty}$ under the operation of $\vee$, so that $\mathbf{b}(\mathbf{y}) = 0$ whenever $\mathbf{X} \cap S(\mathbf{t_y}) = \varnothing$. Similarly, $\mathbf{b}(\mathbf{y}) = \infty$ whenever $\mathbf{X} \cap S_\infty(\mathbf{t_y}) = \varnothing$.

The lattice products $\textcircled{\scriptsize$\vee$}$ and $\textcircled{\scriptsize$\wedge$}$ are called the *multiplicative maximum* and *multiplicative minimum*, respectively. In analogy with Eqs. 4.8.10 and 4.8.11, the *left multiplicative max* and *left multiplicative min* are defined as

$$\mathbf{t} \textcircled{\scriptsize$\vee$} \mathbf{a} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \; : \; \mathbf{b}(\mathbf{y}) = \bigvee_{\mathbf{x} \in \mathbf{X} \cap S_\infty(\mathbf{t}'_\mathbf{y})} [\mathbf{t_x}(\mathbf{y}) \times \mathbf{a}(\mathbf{x})] \, , \; \mathbf{y} \in \mathbf{Y} \right\} \tag{4.8.14}$$

and

$$\mathbf{t} \textcircled{\scriptsize$\wedge$} \mathbf{a} = \left\{ (\mathbf{y}, \mathbf{b}(\mathbf{y})) \; : \; \mathbf{b}(\mathbf{y}) = \bigwedge_{\mathbf{x} \in \mathbf{X} \cap S_\infty(\mathbf{t}'_\mathbf{y})} \left[\mathbf{t_x}(\mathbf{y}) \times' \mathbf{a}(\mathbf{x})\right] \, , \; \mathbf{y} \in \mathbf{Y} \right\} , \tag{4.8.15}$$

respectively. The duality relation between the multiplicative max and min is given by

$$\mathbf{a} \textcircled{\scriptsize$\wedge$} \mathbf{t} = \overline{(\bar{\mathbf{t}} \textcircled{\scriptsize$\vee$} \bar{\mathbf{a}})} \, ,$$

where $\bar{\mathbf{a}}(\mathbf{x}) = \overline{\mathbf{a}(\mathbf{x})}$ and $\bar{\mathbf{t}}_\mathbf{x}(\mathbf{y}) = \overline{[\mathbf{t_y}(\mathbf{x})]}$. Here $\bar{r}$ denotes the additive conjugate of *r* in $\mathbb{R}^{\geq 0}_{\infty}$ and is defined by

$$\bar{r} = \begin{cases} 1/r & \text{if } \; r \in \mathbb{R}^+ \\ 0 & \text{if } \; r = \infty \\ \infty & \text{if } \; r = 0 \, . \end{cases}$$

This is similar to the additive conjugate defined in Eq. 3.12.1 (Section **3.12**).

4.8.4 **Example:** (*Geometric edge filtering.*) Roughly speaking, a local edge in a digital image is a small area in the image where there is a sharp pixel value transition between neighboring pixels. Local edge filtering enhances these transition zones and suppresses values of pixels whose neighboring pixels are of approximately the same value.

The basic idea underlying most simple edge detection techniques involves the computation of the local derivative operator. This example differs from the standard gradient techniques in that it computes differences in local maxima and minima.

Let $\mathbf{t}$ and $\mathbf{s}$ be templates mapping $\mathbb{Z}^2 \to \left(\mathbb{R}^{\geq 0}_{\infty}\right)^{\mathbb{Z}^2}$ and defined by

$$\mathbf{t_y} = \begin{array}{|c|c|c|} \hline 2 & 1 & 2 \\ \hline \end{array} \qquad \mathbf{s_y} = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array}$$

Let $f : \mathbb{Z}^2 \to \mathbb{Z}^2$ be the reflection $f(x_1, x_2) = (x_2, x_1)$ and define $\mathbf{t}'$ and $\mathbf{s}'$ by $\mathbf{t}'_{\mathbf{y}} = \mathbf{t}_{\mathbf{y}} \circ f$ and $\mathbf{s}'_{\mathbf{y}} = \mathbf{s}_{\mathbf{y}} \circ f$, respectively. Then

$$\mathbf{t}'_{\mathbf{y}} = \begin{array}{|c|} \hline 2 \\ \hline 1 \\ \hline 2 \\ \hline \end{array} \qquad\qquad \mathbf{s}'_{\mathbf{y}} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline 1 \\ \hline \end{array}$$

The differences between the local maxima and minima in both the horizontal and vertical directions enhances local edges in these directions while smoothing local areas without sharp edge contrast. The particular algorithm is given by

$$\mathbf{e} := \left\{ \left[(\mathbf{a} \ovee \mathbf{t}) - (\mathbf{a} \owedge \mathbf{s}) \right]^2 + \left[\left(\mathbf{a} \ovee \mathbf{t}'\right) - \left(\mathbf{a} \owedge \mathbf{s}'\right) \right]^2 \right\}^{1/2},$$

where **a** denotes the source image and **e** the edge enhanced image. Figure **4.8.3** illustrates the effects of this technique; the source image **a** is shown on the left and the edge enhanced image **e** on the right.

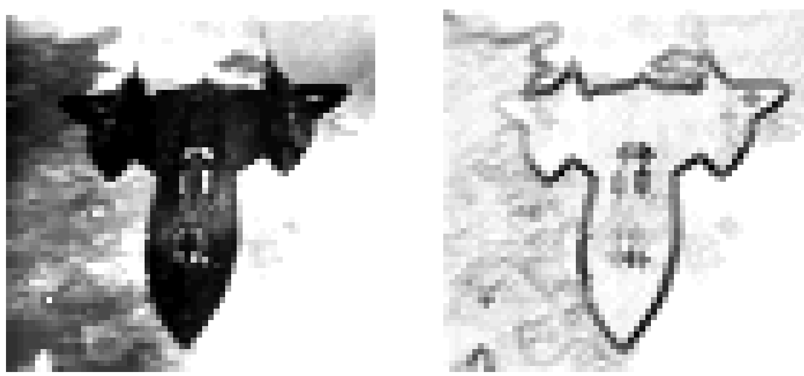

Figure 4.8.3 An example of geometric edge filtering. The source image is on the left and the filtered image on the right.

In both, the definition of the product operator $\oplus$ as well as the four lattice image-template products, we assumed that **X** is a finite point set. There are various instances where this restrictive assumption can be lifted. For example, if **a** is continuous, $\mathbf{t}_{\mathbf{y}}$ is continuous $\forall \mathbf{y} \in \mathbf{Y}$, and $\mathbf{X} \cap S_0(\mathbf{t}_{\mathbf{y}})$ is compact $\forall \mathbf{y} \in \mathbf{Y}$ and the appropriate zero, then $\mathbf{a} \boxed{\vee} \mathbf{t}$, $\mathbf{a} \boxed{\wedge} \mathbf{t}$, $\mathbf{a} \ovee \mathbf{t}$, and $\mathbf{a} \owedge \mathbf{t}$ all exist. Here we assume that the global reduce operations $\bigvee$ and $\bigwedge$ denote the *sup* and *inf* of the functions $\mathbf{a} + \mathbf{t}_{\mathbf{y}}$, $\mathbf{a} +' \mathbf{t}_{\mathbf{y}}$, etc. In case of the product operator $\oplus$, Eq. 4.8.7 assumes the form

$$\mathbf{b}(\mathbf{y}) = \int_{\mathbf{X} \cap S(\mathbf{t}_{\mathbf{y}})} \mathbf{a}(\mathbf{x}) \cdot \mathbf{t}_{\mathbf{y}}(\mathbf{x}) d\mathbf{x}.$$

Extensions to the continuous case are essential for modelling or computing continuous image transformations.

4.9 The Algebra of Templates

Since a template is simply an image whose values are images, the basic elementary function theory concepts that were applied to images are also applied to templates. For example, restrictions and extensions have analogous meaning when applied to templates. If $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ and $\mathbf{Z} \subset \mathbf{Y}$, then $\mathbf{t}|_{\mathbf{Z}}$ denotes the template $\mathbf{s} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Z}}$ defined by $\mathbf{s_y} = \mathbf{t_y} \;\; \forall \mathbf{y} \in \mathbf{Z}$. If $\mathbf{V} \subset \mathbf{X}$, then $\mathbf{t}\|_{\mathbf{V}}$ denotes the template $\mathbf{s} \in \left(\mathbb{F}^{\mathbf{V}}\right)^{\mathbf{Y}}$ defined by $\mathbf{s_y} = \mathbf{t_y}|_{\mathbf{V}} \;\; \forall \mathbf{y} \in \mathbf{Y}$. As in the case of image restrictions, these two concepts of template restrictions can be combined into a single, mutually inclusive definition.

4.9.1 Definition. If $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$, $\mathbf{Z} \subset \mathbf{Y}$, and $\mathbf{V} \subset \mathbf{X}$, then *restriction of* $\mathbf{t}$ *to* $\mathbf{Z}$ *and* $\mathbf{V}$ is defined as

$$\mathbf{t}|_{(\mathbf{Z},\mathbf{V})} = (\mathbf{t}|_{\mathbf{Z}})\|_{\mathbf{V}} .$$

Thus, $\mathbf{t}|_{(\mathbf{Z},\mathbf{X})} = \mathbf{t}|_{\mathbf{Z}}$, $\mathbf{t}|_{(\mathbf{Y},\mathbf{V})} = \mathbf{t}\|_{\mathbf{V}}$, and if $\mathbf{s} = \mathbf{t}|_{(\mathbf{Z},\mathbf{V})}$, then $\mathbf{s} \in \left(\mathbb{F}^{\mathbf{V}}\right)^{\mathbf{Z}}$ and $\mathbf{s_y}(\mathbf{x}) = \mathbf{t_y}(\mathbf{x}) \; \forall \mathbf{y} \in \mathbf{Z}$ and $\forall \mathbf{x} \in \mathbf{V}$.

In order to define the complementary notions of template extensions, let $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ and $\mathbf{r} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{W}}$. The *extension of* $\mathbf{t}$ *to* $\mathbf{r}$ is denoted by $\mathbf{t}|^{\mathbf{r}}$ and is an element of $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}\cup\mathbf{W}}$ defined by

$$\mathbf{t}|^{\mathbf{r}} = \begin{cases} \mathbf{t_y} & \text{if } \mathbf{y} \in \mathbf{Y} \\ \mathbf{r_y} & \text{if } \mathbf{y} \in \mathbf{W}\backslash\mathbf{Y} . \end{cases}$$

On the other hand, if $\mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ and $\mathbf{s} \in \left(\mathbb{F}^{\mathbf{W}}\right)^{\mathbf{Y}}$, then the *range extension of* $\mathbf{t}$ *to* $\mathbf{s}$ is denoted by $\mathbf{t}\|^{\mathbf{s}}$ and defined by $(\mathbf{t}\|^{\mathbf{s}})_{\mathbf{y}} = \mathbf{t_y}|^{\mathbf{s_y}}$. Thus, if $\mathbf{r} = \mathbf{t}\|^{\mathbf{s}}$, then $\mathbf{r} \in \left(\mathbb{F}^{\mathbf{X}\cup\mathbf{W}}\right)^{\mathbf{Y}}$.

The common unary and binary operations on templates correspond to those defined on images. For example, if $f : \mathbb{E} \to \mathbb{F}$ and $\mathbf{t} \in \left(\mathbb{E}^{\mathbf{X}}\right)^{\mathbf{Y}}$, then $\mathbf{r} = f \circ \mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ is defined by

$$\mathbf{r_y} = f(\mathbf{t_y}) ,$$

where f is applied pointwise to the image $\mathbf{t_y}$ as in Eq. 4.7.3. Similarly, if $\gamma : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$, $\mathbf{s} \in \left(\mathbb{E}^{\mathbf{X}}\right)^{\mathbf{Y}}$, and $\mathbf{t} \in \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}}$, then the induced binary operation $\gamma : \left(\mathbb{E}^{\mathbf{X}}\right)^{\mathbf{Y}} \times \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}} \to \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ is defined by

$$(\mathbf{s}\gamma\mathbf{t})_{\mathbf{y}} = \mathbf{s_y}\gamma\mathbf{t_y} \quad \forall \mathbf{y} \in \mathbf{Y} .$$

Thus, if $\mathbb{R} = \mathbb{E} = \mathbb{G} = \mathbb{F}$, and $\mathbf{s}, \mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{Y}}$, then the basic binary operations of addition, multiplication, and maximum are given by

$$\begin{aligned} \mathbf{r_y} &= \mathbf{s_y} + \mathbf{t_y} , \;\; \text{where } \mathbf{r} = \mathbf{s} + \mathbf{t} ; \\ \mathbf{r_y} &= \mathbf{s_y} \cdot \mathbf{t_y} , \;\; \text{where } \mathbf{r} = \mathbf{s} \cdot \mathbf{t} ; \\ \mathbf{r_y} &= \mathbf{s_y} \vee \mathbf{t_y} , \;\; \text{where } \mathbf{r} = \mathbf{s} \vee \mathbf{t} . \end{aligned}$$

4.9.2 **Example:** (*Template arithmetic.*) Suppose $\mathbf{s}, \mathbf{t} \in \left(\mathbb{R}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ are the following translation invariant templates:

$$\mathbf{s_y} = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline \end{array} \qquad \mathbf{t_y} = \begin{array}{|c|} \hline 1 \\ \hline 3 \\ \hline -1 \\ \hline \end{array}$$

Using the basic binary operations of template addition, multiplication, and maximum we obtain the templates

$$(\mathbf{s}+\mathbf{t})_{\mathbf{y}} = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 5 & 1 \\ \hline & -1 & \\ \hline \end{array} \quad (\mathbf{s}\cdot\mathbf{t})_{\mathbf{y}} = \begin{array}{|c|} \hline 6 \\ \hline \end{array} \quad (\mathbf{s}\vee\mathbf{t})_{\mathbf{y}} = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 3 & 1 \\ \hline \end{array}$$

If $(\mathbb{F}, +)$ is a (commutative) group, then according to Theorem **4.4.1** $\left(\mathbb{F}^{\mathbf{X}}, +\right)$ is also a (commutative) group. Replacing the group $\mathbb{F}$ by the induced group $\mathbb{F}^{\mathbf{X}}$ in Theorem **4.4.1** results in the conclusion that $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ is a (commutative) group. This proves the following theorem:

4.9.3 **Theorem.** *If $(\mathbb{F}, +)$ is a (commutative) group, then $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +\right)$ is a (commutative) group.*

If $\mathbf{0}$ denotes the zero image of $\left(\mathbb{F}^{\mathbf{X}}, +\right)$, then the *zero* template of $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +\right)$, denoted by $\emptyset$, is defined by $\emptyset_{\mathbf{y}} = \mathbf{0}\ \ \forall \mathbf{y} \in \mathbf{Y}$. Thus, $\emptyset_{\mathbf{y}}(\mathbf{x}) = 0\ \ \forall \mathbf{x} \in \mathbf{X}$ and $\forall \mathbf{y} \in \mathbf{Y}$, $\mathbf{t} + \emptyset = \emptyset + \mathbf{t} = \mathbf{t}\ \ \forall \mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$, and $S_0(\emptyset_{\mathbf{y}}) = \varnothing\ \ \ \forall \mathbf{y} \in \mathbf{Y}$.

If $(\mathbb{F}, +, \cdot)$ is a ring with multiplicative identity 1, then we can define a multiplicative identity template $\mathbf{l} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ by $\mathbf{l_y} = \mathbf{1}\ \ \forall \mathbf{y} \in \mathbf{Y}$, where $\mathbf{1}$ denotes the multiplicative identity image of $\left(\mathbb{F}^{\mathbf{X}}, +, \cdot\right)$. Thus, $\mathbf{l_y}(\mathbf{x}) = 1\ \ \forall \mathbf{x} \in \mathbf{X}$ and $\forall \mathbf{y} \in \mathbf{Y}$, $\mathbf{t} \cdot \mathbf{l} = \mathbf{l} \cdot \mathbf{t} = \mathbf{t}\ \ \forall \mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$, and $S_0(\mathbf{l_y}) = \mathbf{X}\ \ \forall \mathbf{y} \in \mathbf{Y}$. The next theorem is an easy consequence of Theorem **4.4.2**.

4.9.4 **Theorem.** *If $(\mathbb{F}, +, \cdot)$ is a (commutative) ring (with unity), then $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +, \cdot\right)$ is a (commutative) ring (with unity).*

As can be inferred from the last two theorems, the algebra of templates behaves very much like the algebra of the underlying value set. In view of theorems **4.4.1** and **4.4.2**, and the fact that templates are images, this should come as no great surprise. Of course, just as for images in general, the induced structure $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +, \cdot\right)$ is, in general, weaker than the original structure $(\mathbb{F}, +, \cdot)$. For example, it easily follows from our earlier observations concerning the algebraic structure of $\mathbb{F}^{\mathbf{X}}$ that if $(\mathbb{F}, +, \cdot)$ is a division ring, then $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +, \cdot\right)$ is a von Neumann ring with zero divisors and, hence, not a division ring.

There is a natural connection between the algebra of $\mathbb{F}$–valued templates and the algebra of matrices with entries from $\mathbb{F}$. Suppose that $\mathbf{X}$ and $\mathbf{Y}$ are finite, say $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ and $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$. Then there is a natural map

$$\begin{aligned} \psi : \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}} &\to \mathrm{M}_{mn}(\mathbb{F}) \\ \mathbf{t} &\rightarrowtail (t_{ij})_{m\times n} \end{aligned} \tag{4.9.1}$$

defined by $t_{ij} = \mathbf{t}_{\mathbf{y}_j}(\mathbf{x}_i)$. Another way of visualizing the map ψ is by using the image-to-vector map

$$\begin{aligned} \nu : \mathbb{F}^{\mathbf{X}} &\to \mathbb{F}^m \\ \mathbf{a} &\rightarrowtail \nu(\mathbf{a}) \end{aligned}$$

defined by $\nu(\mathbf{a}) = (\mathbf{a}(\mathbf{x}_1), \ldots, \mathbf{a}(\mathbf{x}_m))$. Then $\psi(\mathbf{t}) = \left(\nu(\mathbf{t}_{\mathbf{y}_1})', \ldots, \nu(\mathbf{t}_{\mathbf{y}_n})'\right)$, where $\nu\left(\mathbf{t}_{\mathbf{y}_j}\right)'$ denotes the transpose of the vector $\nu\left(\mathbf{t}_{\mathbf{y}_j}\right) = \left(\mathbf{t}_{\mathbf{y}_j}(\mathbf{x}_1), \ldots, \mathbf{t}_{\mathbf{y}_j}(\mathbf{x}_m)\right)$.

4.9.5 **Example:** Suppose $\mathbf{s} \in \left(\mathbb{R}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ is defined by

$$\mathbf{s}_{\mathbf{y}} = \begin{array}{|c|c|c|} \hline & 1 & \\ \hline 1 & 1 & 1 \\ \hline & 1 & \\ \hline \end{array}$$

Let $\mathbf{X} = \left\{(x_1, x_2) \in \mathbb{Z}^2 : 1 \le x_1 \le 3,\ 1 \le x_2 \le 4\right\}$ and $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{X}}$ be the restriction of $\mathbf{s}$ to the point set $\mathbf{X}$ in both its domain and range. More precisely, $\mathbf{t} = \mathbf{s}|_{(\mathbf{X}, \mathbb{R}^{\mathbf{X}})}$ so that $\mathbf{t}_{\mathbf{y}}(\mathbf{x}) = \mathbf{s}_{\mathbf{y}}(\mathbf{x})\ \forall\, \mathbf{y}, \mathbf{x} \in \mathbf{X}$. In particular, the image corresponding to $\mathbf{t}_{(1,1)}$ has the following appearance:

$$\mathbf{t}_{(1,1)} = \begin{array}{|c|c|c|c|} \hline 1 & 1 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array}$$

The region outlined in bold show the support of **t**. If we use the usual row scanning order $\mathbf{x}_k = (x_i, x_j) \Leftrightarrow k = 4(i-1) + j$ for relabeling the array $\mathbf{X}$, then the vector corresponding to the image $\mathbf{t}_{(1,1)}$ is given by

$$\begin{aligned} \nu(\mathbf{t}_{\mathbf{x}_1}) &= (\mathbf{t}_{\mathbf{x}_1}(1,1), \mathbf{t}_{\mathbf{x}_1}(1,2), \ldots, \mathbf{t}_{\mathbf{x}_1}(3,4)) \\ &= (1,1,0,0,1,0,0,0,0,0,0,0)\,. \end{aligned}$$

By definition, this vector forms the first column of the matrix $\psi(\mathbf{t}) = \left(\nu(\mathbf{t}_{\mathbf{x}_1})', \ldots, \nu(\mathbf{t}_{\mathbf{x}_{12}})'\right)$. Similarly, the second column is derived from the vector

$$\begin{aligned}\nu(\mathbf{t}_{\mathbf{x}_2}) &= (\mathbf{t}_{\mathbf{x}_2}(1,1), \mathbf{t}_{\mathbf{x}_2}(1,2), \ldots, \mathbf{t}_{\mathbf{x}_2}(3,4)) \\ &= (1,1,1,0,0,1,0,0,0,0,0,0).\end{aligned}$$

where the image $\mathbf{t}_{\mathbf{x}_2} = \mathbf{t}_{(1,2)}$ is given by

$\mathbf{t}_{(1,2)} =$

1	1	1	0
0	1	0	0
0	0	0	0

Continuing in this fashion, we obtain the symmetric matrix

$$\psi(\mathbf{t}) = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

The relationship between matrix algebra and the algebra of templates is given by the following theorem.

4.9.6 **Theorem.** $\psi : \left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, +, \cdot\right) \to (\mathrm{M}_{mn}(\mathbb{F}), +, \cdot)$ *is an isomorphism.*

Proof: To show that ψ is one-to-one, let $\mathbf{s}, \mathbf{t} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ with $\mathbf{s} \neq \mathbf{t}$. Then $\exists \mathbf{y}_j \in \mathbf{Y}$ such that $\mathbf{t}_{\mathbf{y}_j} \neq \mathbf{s}_{\mathbf{y}_j}$. But this means that $\exists i$ such that $\mathbf{t}_{\mathbf{y}_j}(\mathbf{x}_i) \neq \mathbf{s}_{\mathbf{y}_j}(\mathbf{x}_i)$ and, therefore, $t_{ij} \neq s_{ij}$ for some pair of indices *i*, *j*. Thus, $\psi(\mathbf{t}) \neq \psi(\mathbf{s})$.
For $\mathrm{U} = (u_{ij})_{m \times n} \in \mathrm{M}_{mn}(\mathbb{F})$, define $\mathbf{u} \in \left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ by $\mathbf{u}_{\mathbf{y}_j}(\mathbf{x}_i) = u_{ij}$. Then $\psi(\mathbf{u}) = \mathrm{U}$. This proves that ψ is onto.

If $\mathbf{r} = \mathbf{s} + \mathbf{t}$, $\psi(\mathbf{s}) = \mathrm{S} = (s_{ij})_{m \times n}$, $\psi(\mathbf{t}) = \mathrm{T} = (t_{ij})_{m \times n}$, and $\mathrm{S} + \mathrm{T} = \mathrm{R} = (r_{ij})_{m \times n}$, then

$$\begin{aligned}\mathbf{r}_{\mathbf{y}_j}(\mathbf{x}_i) &= \mathbf{s}_{\mathbf{y}_j}(\mathbf{x}_i) + \mathbf{t}_{\mathbf{y}_j}(\mathbf{x}_i) = s_{ij} + t_{ij} \\ &= r_{ij} \quad \forall i \in \{1, \ldots, m\} \text{ and } \forall j \in \{1, \ldots, n\}.\end{aligned}$$

Therefore, $\psi(\mathbf{s}) + \psi(\mathbf{t}) = \mathrm{S} + \mathrm{T} = \mathrm{R} = \psi(\mathbf{r}) = \psi(\mathbf{s} + \mathbf{t})$.

Similarly, $\psi(\mathbf{s}) \cdot \psi(\mathbf{t}) = \psi(\mathbf{s} \cdot \mathbf{t})$.

Q.E.D.

This shows that for finite point sets, templates are equivalent to matrices and the induced binary operations on $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ are equivalent to the induced binary operations on $\mathrm{M}_{mn}(\mathbb{F})$; e.g. template *addition* corresponds to matrix *addition*, and template *multiplication* corresponds to Hadamard (pointwise) matrix *multiplication*.

4.10 Template Products

According to Theorem **4.9.6**, if $(\mathbb{F}, \gamma, \bigcirc)$ is a ring (or semiring), then $\left(\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}, \gamma, \bigcirc\right)$ is also a ring (or semiring). The operations γ and $\bigcirc$ on $\left(\mathbb{F}^{\mathbf{X}}\right)^{\mathbf{Y}}$ are pointwise induced operations. In Section **4.8** we observed that under certain conditions the operations γ and $\bigcirc$ can be combined to induce the more complex image–template product operator $\textcircled{\gamma}$. This notion extends to templates as well.

Suppose $\mathbf{s} \in \left(\mathbb{E}^{\mathbf{Z}}\right)^{\mathbf{X}}$, $\mathbf{t} \in \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}}$, $\bigcirc : \mathbb{E} \times \mathbb{G} \to \mathbb{F}$, $(\mathbb{F}, \gamma)$ a commutative semigroup, and **X** a finite point set. Then the template product $\mathbf{r} = \mathbf{s} \textcircled{\gamma} \mathbf{t}$, where $\mathbf{r} \in \left(\mathbb{F}^{\mathbf{Z}}\right)^{\mathbf{Y}}$, is defined as

$$\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = \mathop{\Gamma}_{\mathbf{x} \in \mathbf{X}} \left(\mathbf{s}_{\mathbf{x}}(\mathbf{z}) \bigcirc \mathbf{t}_{\mathbf{y}}(\mathbf{x})\right), \quad \forall \mathbf{y} \in \mathbf{Y} \text{ and } \forall \mathbf{z} \in \mathbf{Z}. \tag{4.10.1}$$

By definition, the operator $\textcircled{\gamma}$ is a function

$$\textcircled{\gamma} : \left(\mathbb{E}^{\mathbf{Z}}\right)^{\mathbf{X}} \times \left(\mathbb{G}^{\mathbf{X}}\right)^{\mathbf{Y}} \to \left(\mathbb{F}^{\mathbf{Z}}\right)^{\mathbf{Y}}.$$

Pictorially we can view $\mathbf{r} = \mathbf{s} \textcircled{\gamma} \mathbf{t}$ as a functional composition with **t** applied as a $\mathbb{G}$–valued template from **Y** to **X**, followed by the $\mathbb{E}$–valued template **s** from **X** to **Z** as shown in Figure **4.10.1**.

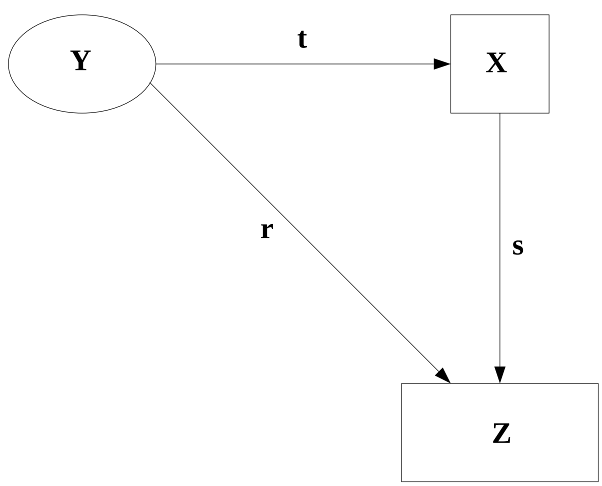

Figure 4.10.1 Illustration of template composition.

Similar to the computation of the image–template product, it is generally not necessary to apply the global reduce operator Γ in Eq. 4.10.1 to all of **X** but only over a certain subset of **X**. For illustration purposes we again use the primary value sets $\mathbb{R}$, $\mathbb{R}_{\pm\infty}$, and $\mathbb{R}_{\infty}^{\geq 0}$.

If $\mathbf{s} \in \left(\mathbb{R}^{\mathbf{Z}}\right)^{\mathbf{X}}$ and $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}\right)^{\mathbf{Y}}$, then for a given $\mathbf{y} \in \mathbf{Y}$ define

$$S_{\mathbf{y}}(\mathbf{z}) = \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{z} \in S(\mathbf{s}_{\mathbf{x}}) \text{ and } \mathbf{x} \in \mathrm{S}(\mathbf{t}_{\mathbf{y}})\} \;\; \forall \mathbf{z} \in \mathbf{Z}\,. \tag{4.10.2}$$

It follows that $\mathbf{s}_{\mathbf{x}}(\mathbf{z}) \cdot \mathbf{t}_{\mathbf{y}}(\mathbf{x}) = 0$ whenever $\mathbf{x} \notin S_{\mathbf{y}}(\mathbf{z})$. Hence, in order to compute $\mathbf{r} = \mathbf{s} \oplus \mathbf{t}$, we can use the formula

$$\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = \sum_{\mathbf{x} \in S_{\mathbf{y}}(\mathbf{z})} \mathbf{s}_{\mathbf{x}}(\mathbf{z}) \cdot \mathbf{t}_{\mathbf{y}}(\mathbf{x})\,, \tag{4.10.3}$$

where $\sum\limits_{\mathbf{x} \in S_{\mathbf{y}}(\mathbf{z})} \mathbf{s}_{\mathbf{x}}(\mathbf{z}) \cdot \mathbf{t}_{\mathbf{y}}(\mathbf{x}) \equiv 0$ whenever $S_{\mathbf{y}}(\mathbf{z}) = \varnothing$.

The lattice product $\mathbf{r} = \mathbf{s} \boxed{\vee}\, \mathbf{t}$ is defined in a similar manner. For $\mathbf{s} \in \left(\mathbb{R}^{\mathbf{Z}}_{\pm\infty}\right)^{\mathbf{X}}$, $\mathbf{t} \in \left(\mathbb{R}^{\mathbf{X}}_{\pm\infty}\right)^{\mathbf{Y}}$ and $\mathbf{y} \in \mathbf{Y}$ define

$$S^{-\infty}_{\mathbf{y}}(\mathbf{z}) = \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{z} \in S_{-\infty}(\mathbf{s}_{\mathbf{x}}) \text{ and } \mathbf{x} \in \mathrm{S}_{-\infty}(\mathbf{t}_{\mathbf{y}})\} \;\; \forall \mathbf{z} \in \mathbf{Z}\,. \tag{4.10.4}$$

Then $\mathbf{r} \in \left(\mathbb{R}^{\mathbf{Z}}_{\pm\infty}\right)^{\mathbf{Y}}$ can be defined using the formula:

$$\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = \bigvee_{\mathbf{x} \in S^{-\infty}_{\mathbf{y}}(\mathbf{z})} [\mathbf{s}_{\mathbf{x}}(\mathbf{z}) + \mathbf{t}_{\mathbf{y}}(\mathbf{x})]\,, \tag{4.10.5}$$

where $\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = -\infty$ whenever $S^{-\infty}_{\mathbf{y}}(\mathbf{z}) = \varnothing$.

The dual operation $\mathbf{u} = \mathbf{s} \boxed{\wedge}\, \mathbf{t}$ can be computed in terms of the formula

$$\mathbf{u}_{\mathbf{y}}(\mathbf{z}) = \bigwedge_{\mathbf{x} \in S^{\infty}_{\mathbf{y}}(\mathbf{z})} [\mathbf{s}_{\mathbf{x}}(\mathbf{z}) +' \mathbf{t}_{\mathbf{y}}(\mathbf{x})]\,, \tag{4.10.6}$$

where $S^{\infty}_{\mathbf{y}}(\mathbf{z}) = \{\mathbf{x} \in \mathbf{X} \,:\, \mathbf{z} \in S_{\infty}(\mathbf{s}_{\mathbf{x}}) \text{ and } \mathbf{x} \in \mathrm{S}_{\infty}(\mathbf{t}_{\mathbf{y}})\} \;\; \forall \mathbf{z} \in \mathbf{Z}$.

It follows from Eqs. 4.10.5 and 4.10.6 that the duality relation

$$\mathbf{s} \boxed{\wedge}\, \mathbf{t} = (\mathbf{t}^* \boxed{\vee}\, \mathbf{s}^*)^*$$

is preserved. It is also not difficult to verify that the support of $\mathbf{r}_{\mathbf{y}}$, where $\mathbf{r} = \mathbf{s} \oplus \mathbf{t}$ or $\mathbf{r} = \mathbf{s} \boxed{\vee}\, \mathbf{t}$, is given by $S(\mathbf{r}_{\mathbf{y}}) = \{\mathbf{z} \in \mathbf{Z} \,:\, S_{\mathbf{y}}(\mathbf{z}) \neq \varnothing\}$ or $S_{-\infty}(\mathbf{r}_{\mathbf{y}}) = \left\{\mathbf{z} \in \mathbf{Z} \,:\, S^{-\infty}_{\mathbf{y}}(\mathbf{z}) \neq \varnothing\right\}$, respectively. Similarly, if $\mathbf{u} = \mathbf{s} \boxed{\wedge}\, \mathbf{t}$, then $S_{\infty}(\mathbf{u}_{\mathbf{y}}) = \left\{\mathbf{z} \in \mathbf{Z} \,:\, S^{\infty}_{\mathbf{y}}(\mathbf{z}) \neq \varnothing\right\}$.

For $\mathbf{s} \in \left[\left(\mathbb{R}^{\geq 0}_{\infty}\right)^{\mathbf{Z}}\right]^{\mathbf{X}}$ and $\mathbf{t} \in \left[\left(\mathbb{R}^{\geq 0}_{\infty}\right)^{\mathbf{X}}\right]^{\mathbf{Y}}$, the lattice product $\mathbf{r} = \mathbf{s} \circledvee \mathbf{t}$ is defined by

$$\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = \bigvee_{\mathbf{x} \in S_{\mathbf{y}}(\mathbf{z})} [\mathbf{s}_{\mathbf{x}}(\mathbf{z}) \times \mathbf{t}_{\mathbf{y}}(\mathbf{x})]\,, \tag{4.10.7}$$

where $S_{\mathbf{y}}(\mathbf{z})$ is as in Eq. 4.10.2 and $\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = 0$ whenever $S_{\mathbf{y}}(\mathbf{z}) = \varnothing$. The dual operator $\circledwedge$ can be defined either in terms of duality by

$$\mathbf{s} \circledwedge \mathbf{t} = \overline{(\bar{\mathbf{t}} \circledvee \bar{\mathbf{s}})}\,,$$

or by the formula

$$\mathbf{r}_{\mathbf{y}}(\mathbf{z}) = \bigwedge_{\mathbf{x} \in S^{\infty}_{\mathbf{y}}(\mathbf{z})} [\mathbf{s}_{\mathbf{x}}(\mathbf{z}) \times' \mathbf{t}_{\mathbf{y}}(\mathbf{x})]\,, \tag{4.10.8}$$

where $\mathbf{r} = \mathbf{s} \circledwedge \mathbf{t}$.

For many templates an easy way of computing the template product $\mathbf{r} = \mathbf{s} \circledcirc \mathbf{t}$ is to use the following fact:

4.10.1 **Theorem.** *If* $\mathbf{r} = \mathbf{s} \textcircled{\gamma} \mathbf{t}$, *then* $\mathbf{r_y} = \mathbf{s} \textcircled{\gamma} \mathbf{t_y}$ *and* $\mathbf{r_y}(\mathbf{z}) = \mathbf{s}'_{\mathbf{z}} \bullet \mathbf{t_y}$.

The theorem follows from the observation

$$\mathbf{r_y}(\mathbf{z}) = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{s}'_{\mathbf{z}}(\mathbf{x}) \cdot \mathbf{t_y}(\mathbf{x}) = \sum_{\mathbf{x} \in \mathbf{X}} \mathbf{s_x}(\mathbf{z}) \cdot \mathbf{t_y}(\mathbf{x}) = (\mathbf{s} \oplus \mathbf{t_y})(\mathbf{z}),$$

and

$$\sum_{\mathbf{x} \in \mathbf{X}} \mathbf{s}'_{\mathbf{z}}(\mathbf{x}) \cdot \mathbf{t_y}(\mathbf{x}) = \mathbf{s}'_{\mathbf{z}} \bullet \mathbf{t_y} .$$

There is nothing special about using the operator $\oplus$, the argument is identical when using the general product $\textcircled{\gamma}$.

4.10.2 **Example:** Suppose $\mathbf{s}, \mathbf{t} \in \left(\mathbb{R}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ are the following translation invariant templates:

$\mathbf{s_y}$ =

1	2	1

$\mathbf{t_y}$ =

1
3
-1

Then the template product $\mathbf{r} = \mathbf{s} \oplus \mathbf{t}$ is the template defined by

$\mathbf{r_y}$ =

1	2	1
3	6	3
-1	-2	-1

If $\mathbf{s}, \mathbf{t} \in \left(\mathbb{R}_{\pm\infty}^{\mathbb{Z}^2}\right)^{\mathbb{Z}^2}$ are defined as above with values $-\infty$ outside the support, then the template product $\mathbf{r} = \mathbf{s} \boxed{\vee} \mathbf{t}$ is the template defined by

$\mathbf{r_y}$ =

2	3	2
4	5	4
0	1	0

In this particular example, the templates $\mathbf{r} = \mathbf{s} \boxed{\vee} \mathbf{t}$ and $\mathbf{u} = \mathbf{s} \boxed{\wedge} \mathbf{t}$ have the property that $S_{-\infty}(\mathbf{r_y}) = S_{\infty}(\mathbf{u_y}) \; \forall \mathbf{y} \in \mathbb{Z}^2$ and

$$\mathbf{u_y}(\mathbf{z}) = \begin{cases} \mathbf{r_y}(\mathbf{z}) & \text{if } \mathbf{z} \in S_{\infty}(\mathbf{u_y}) \\ \infty & \text{otherwise} . \end{cases}$$